2017

Computational Studies on Fluid and Particle Dynamics

Chenguang Zhang
Louisiana State University and Agricultural and Mechanical College, czhan21@lsu.edu

COMPUTATIONAL STUDIES ON FLUID AND PARTICLE DYNAMICS

A Dissertation

Submitted to the Graduate Faculty of the
Louisiana State University and
Agricultural and Mechanical College
in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy

in

The College of Engineering

by
Chenguang Zhang
BS, Physical Geography, Nanjing University, 2009
MS, Physical Oceanography and Coastal Sciences, Louisiana State University, 2012
MS, Computer Science, Louisiana State University, 2017
August 2017

Acknowledgments

I would like to thank my advisor, Dr. Krishinaswamy Nandakumar for his guidance, enthusiasm, and insight. Those valuable lessons I learned from him will stay with me forever.

I would like to thank Dr. Harris Wong for his devotion of time and experience, and for bringing my analytical skills to a new level.

I am grateful to Dr. Chunliang Wu, with whom I have worked closely through my PhD, for the countless discussions and lessons on particle dynamics and its simulation methods.

I thank my group members, Dr. Mayur Sathe, Dr. Shivkumar Bale, Dr. Yuehao Li, Dr. Abhijit Rao, and fellow graduate students Gongqiang, Jielie, Daniel, Zhizhong, Aaron, LC, Sai and Sharareh.

Finally, I dedicate this dissertation to my wife, Xiaoning Hou, and my family back in China—I can not imagine how far I could go without their love.

Table of Contents

List of Tables

List of Figures

Abstract

This dissertation is divided into two parts. The first part includes chapter 2 to 4, which focus on development and application of numerical algorithm on particle and fluid simulation. Starting with a pure granular system in a driven cavity setup (Chapter 2), we move on to the immersed boundary simulation of fluid solid interaction (Chapter 3). This part ends with a coupled immersed boundary-discrete particle implementation. The second part includes Chapter 5 and Appendix A, each deals with an independent problem and focuses more on the theoretical aspects. Chapter 5 deals with a classic fluid dynamics problem of thermal wave induced net flux. Appendix A studies the underestimation of solver accuracy when the solution has singularities, and analyzes the cause of the underestimation using a simple theoretical model.

Chapter 1
General Introduction

1.1 Introduction

Granular material refers to materials consist of a large number of microscopic solid grains. It is widely occurring both in nature (sand, snow, etc), and in many industries, including agricultural, civil, chemical, pharmaceutical, etc. In fact it is the second most used material in industry [Richard et al., 2005] (The first one being water).

Granular material is unique in its ability to behave like gas, liquid or solid. Extensive information could be found in the review articles [Jaeger et al., 1996, Kadanoff, 1999, Campbell, 2006] or books [Ristow, 2000, Duran, 2012, Andreotti et al., 2013]. Unlike the dilute or the quasi-static limit which are relatively well understood from lessons in kinetic theory [Brilliantov and Pöschel, 2010] or solid mechanics [Terzaghi, 1943], respectively, the liquid or the "dense flow" regime of granular material is less understood. Valuable lessons are learned by putting granular materials in setups similar to those originally for studying fluid dynamics. Many experiments, such as hydraulic jump [Brennen et al., 1983, Johnson and Gray, 2011], dam break [Lajeunesse et al., 2004, Staron and Hinch, 2005], impact ejecta [Marston et al., 2012], etc are inspired by their classic liquid counterpart. For a similar reason, we introduced a granular lid-driven cavity setup in Chapter 2 and study it numerically using the discrete element method, or DEM.

DEM is the main *numerical* method to study granular material. It was first developed in the 1970s by Cundall and Strack [1979]. It is instructive to compare DEM with molecular dynamics, or MD [Rahman, 1964] that was developed before it. Both methods solve the motion of a number of "molecules" or "elements", which evolve in space according to their mutual interactions and external forcing from the environment. DEM is more complicated than MD in the sense that it needs to handle irregular particle shapes, the rotational degree of freedom, and model the effect of direct contacts.

The governing equation for every element in DEM is just rigid body dynamics

$$m_i \dot{\boldsymbol{u}}_i = \sum_{j \in nbr} (\boldsymbol{F}_{n,ij} + \boldsymbol{F}_{t,ij}) + m_i \boldsymbol{g}$$
$$\mathbb{I}_i \dot{\boldsymbol{\omega}}_i = \sum_{j \in nbr} (\boldsymbol{r} \times \boldsymbol{F}_{t,ij})$$

where subscript i is an index used for particle identity, m and $\mathbb{I}$ are the mass and moment of inertia tensor, g is the gravity. The force in the parenthesis is due to direct contact with a neighboring (nbr) particle j. It is decomposed into normal ($\boldsymbol{F}_n$) and tangential ($\boldsymbol{F}_t$) components as the interaction mechanisms in those two directions are quite distinct and need to be modeled differently. The Hertz-Mindline model is used in Chapter 2. Its detailed expression could be found in Wu et al. [2014].

DEM was originally developed to solve civil engineering problems like soil or rock dynamics. Compared with the traditional continuum based method, DEM resolves the discrete nature of its components that are important in certain phenomena like shear band/fracture, where large deformation or topological change happens. DEM nowadays applies to a much wider range of problems involving granular flows, and has been extended to include heat transfer, interaction with structure or fluids.

In reality, granular materials exist mostly in an ambient fluid like air or water. In many scenarios the fluid plays important roles in the dynamics of the granular system, common examples include sedimentation in the natural environment, and the fluidized bed in industry. The DEM method discussed before has been coupled together with well established computational fluid dynamics (CFD) algorithms to simulate those gas-solid or liquid-solid system. The solid-solid interaction is processed in the DEM framework, while the fluid dynamics is processed by CFD method. The method of coupling the two differs according to the level of spatial resolutions of the solid phase. In Chapter 3, we developed and validated our implementation of the immersed boundary method (IBM), which itself is a kind of direct numerical simulation (DNS) method of fluid-solid systems.

The main contribution is a convenient solid shape representation using the signed distance field (SDF). This representation enables an accurate volume fraction calculation without the complexity of intersection tests. In addition, complex geometries can be formed with ease through the Boolean properties of SDF.

The DNS way of simulating fluid-solid system is limited by its high computational cost. Since the fluid field around each particles has to be resolved, the mesh size is very high even for a few thousand particles at moderate Reynolds number. For this reason, the discrete particle method (DPM) is used in Chapter 4. DPM, as a modeling framework, uses DEM as a sub component. It differs from the IBM in Chapter 3 mainly in the level of solid resolution. In DPM the solid size is smaller than the CFD mesh size so the fluid field around each solid is not resolved. Instead, the effects of fluid on solid is accounted mainly by a pressure term and an empirical "drag coefficient" in the drag force ($\boldsymbol{F}_d$). In other words, the linear momentum equation in DEM now becomes

$$m_i \dot{\boldsymbol{u}}_i = -V_i \nabla p + \boldsymbol{F}_d + \sum_{j \in nbr} (\boldsymbol{F}_{n,ij} + \boldsymbol{F}_{t,ij}) + m_i \boldsymbol{g}.$$

V is the particle volume, and p is fluid pressure. Other effects like the Basset force, added mass, Magnus force are often less important, but could be included if necessary. Chapter 4 uses an existing in-house DPM implementation by Dr. Chunliang Wu [Wu et al., 2009] and couples it with the immersed boundary method in Chapter 3. A two dimensional version of the granular impact experiment is simulated. The simulation result is quite similar to the cited experiment.

Chapter 5 and 6 each focuses on a theme different from previous chapters. Chapter 5 is a revisit of the classic thermal wave problem, which studies the generation of net flux by transient heating at the boundary. After validating the linear theory by simulation, we apply asymptotic analysis to study the limiting behavior of the flux on the thermal wave length. The parameter combinations that maximize the flux are explored in detail. In

Chapter 6 we study the underestimation of a partial differential equation (PDE) solver's *spatial* accuracy when the solution itself has singularities. We found the singularity-caused underestimation exists not just for incompressible flow solver, but also for a simple Laplacian equation solver and the even simpler one dimensional quadrature with the rectangle rule. We explain the cause of the underestimation using a simple theoretical model.

Chapter 2
Discrete Element Study in a 2D Granular Cavity

2.1 Introduction

This chapter presents a study of a 2D granular cavity setup illustrated in Figure 2.1. A number of mono-dispersed particles are aligned as a single layer (hence 2D). They are

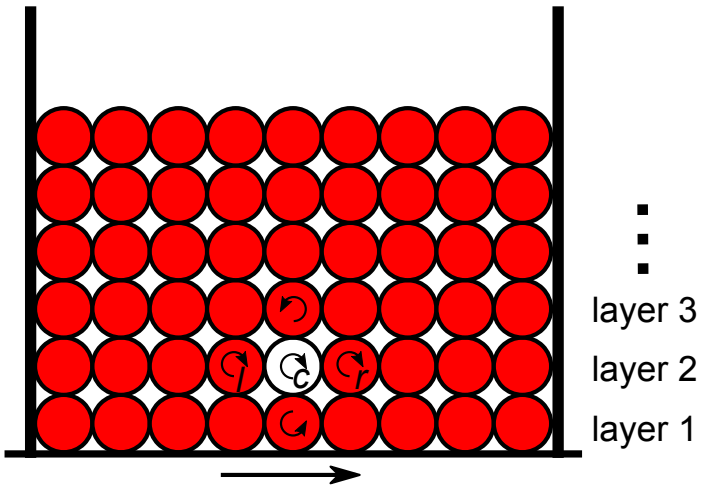

Figure 2.1: Illustration of the 2D cavity setup in this chapter.

bounded from the left and right sides by two vertical walls, and sit on a horizontal belt that slides at a constant speed. This study originates from a lab experiment of granular cavity to study persistent lid-driven circulations, which was indeed observed when a wide (in depth direction) cavity is used. With a narrower cavity that can only fit one layer in the depth direction, we observed that the system of particles alternates between a "jamming" state which is essentially motionless, and a "flowing" state, where the particles circulate. The original purpose of using a narrower (in depth direction) cavity is, by the way, for easier data collection as there is no need to consider the depth-wise motion. We will present the study using a wider cavity elsewhere in the future, and focus in this chapter on a discrete element study of an ideal 2D cavity, which align particles perfectly as a 2D layer without touching front or back walls.

This setup is very similar to the liquid lid driven cavity with two differences besides using particles instead of liquid. The first is particles are driven from below—instead of top—to maintain contacts with the belt, this difference is only superficial as the liquid cavity can be flipped vertically without altering its dynamics. The other and more essential difference is the granular cavity has a free surface. The reason is in case a top lid is used, we have a finite container and introduce a new variable of the particle filling fraction. It is however not straightforward to choose a proper value for it. On one hand, a low filling fraction (a small amount of particles in a large cavity) is no different from the free surface setup; on the other hand, with a high filling fraction there is limited space for Reynolds dilation [Reynolds, 1886] and particles won't be able to move much. We use free surface to avoid this complexity.

2.2 Method and Validation

2.2.1 DEM Method

The DEM method used by us was presented in detail in Wu et al. [2014]. The motion of each particle follows the rigid body dynamics, and only the calculation of the contact force is summarized here, which was based on [Tsuji et al., 1992, Shäfer et al., 1996]. Considering two moving *spherical* particles i and j of the same radius R, with position, velocity and angular velocity represented by symbols $\boldsymbol{x}$, $\boldsymbol{v}$, $\boldsymbol{\omega}$, respectively. The normal force is calculated by $(-k_n\delta_n^{3/2} - \eta_n \boldsymbol{v}_{ij}\cdot\boldsymbol{n}_{ij})\boldsymbol{n}_{ij}$, where $\delta = R_i + R_j - |\boldsymbol{x}_j - \boldsymbol{x}_i|$ is the overlap of the two spheres, $\boldsymbol{v}_{ij} = \boldsymbol{v}_i - \boldsymbol{v}_j$ is their relative velocities, and

$$\boldsymbol{n}_{ij} = \frac{\boldsymbol{x}_j - \boldsymbol{x}_i}{|\boldsymbol{x}_j - \boldsymbol{x}_i|},$$

is the normal direction at the contact point from particle i to j. The coefficients are

$$k_n = \frac{E_s\sqrt{2R}}{3(1-\sigma_s^2)}, \eta_n = 2\sqrt{mk_n}\frac{|\ln n|}{\pi^2 + \ln^2 n}$$

n is the normal restitution coefficient, E_s, $G_s = E_s/(2(1+\sigma_s))$ are the Young and shear modules of the particle material, respectively, and σ_s is the Poisson ratio. The tangential force is calculated by $-k_t\boldsymbol{\delta}_t - \eta_t \boldsymbol{v}_{t,ij}$, where

$$\boldsymbol{\delta}_t = -\boldsymbol{n}_{ij} \times (\boldsymbol{n}_{ij} \times \boldsymbol{\delta}_t^0) + \boldsymbol{v}_{t,ij} dt$$

is the tangential displacement calculated as a time integration, with $\boldsymbol{\delta}_t^0$ the value at the previous time step. $\boldsymbol{v}_{t,ij} = \boldsymbol{v}_{ij} - (\boldsymbol{v}_{ij} \cdot \boldsymbol{n}_{ij})\boldsymbol{n}_{ij} + (R_i\boldsymbol{\omega}_i + R_j\boldsymbol{\omega}_j) \times \boldsymbol{n}_{ij}$ is the tangential relative velocity at the point of contact. The coefficients are

$$k_t = \frac{2\sqrt{2R}G_s}{2-\sigma_s}\sqrt{\delta}, \eta_t = 0.5\eta_n$$

2.2.2 Validation by Collapse of a 2D Granular Column

Consider a ground plane at $y = 0$, and a rectangular *lattice* of granular material in $[-R_0, R_0] \times [0, \alpha R_0]$, where α is the initial aspect ratio. By giving the grains a minor perturbation, the column collapses under gravity. The final column shape after all particles stop moving has a half width, or a "run-out distance" of R_∞ which, after nondimensionalized by the initial column width R_0, can be correlated with the initial aspect ratio α. The lab experiment of a 3D cylindrical column was first done by Lajeunesse [2004] and simulated in 2D through contact dynamics by Staron and Hinch [2005]. We compare the simulation result from our code against the correlation in [Staron and Hinch, 2005] section 4.2. The initial column half width is fixed as $R_0 = 12.5mm$, and we control the initial aspect ratio α through the column height. The particles are given initial horizontal velocity of $1mm/s$ as a perturbation. Particle parameters are listed in Table 2.1. To measure the final width of the column, we sort the particles' x-position and discard the first and last 20 particles to reject those loose particles that roll to further distance. Staron and Hinch [2005] uses the concept of connected component for the same purpose. We experimented by rejecting different number of particles and found this simple measurement is quite robust. Figure 2.2

Table 2.1: Parameters for granular column collapse validation

Parameter	Value
Particle diameter	1 mm
Particle density	2500 kg/m3
Young's modulus	2×10^8
Normal restitution coeff. p-p, p-w	0.65, 0.65
Friction coefficient p-p, p-w	0.9, 0.9
Time step	$1 \times 10^{-6}s$

compares the simulation result with the correlation in [Staron and Hinch, 2005] and they agree very well.

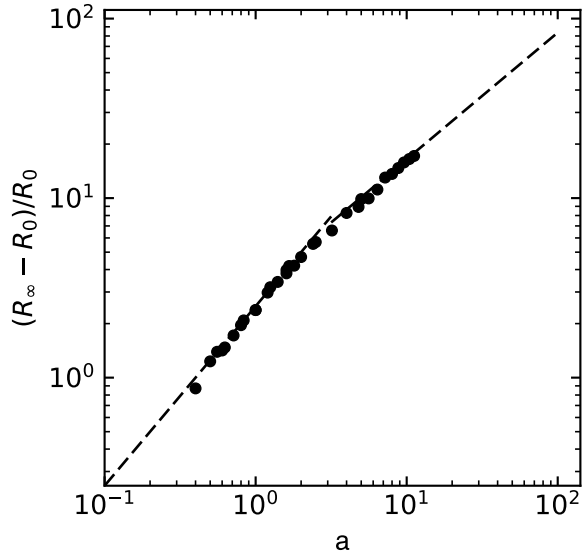

Figure 2.2: Comparison between the simulation result (dot) and correlation (dashed) published in [Staron and Hinch, 2005]. y axis is the nondimensionalized column width $(R_\infty - R_0)/R_0$, and x axis is the initial aspect ratio α. The two correlation are $(R_\infty - R_0)/R_0 = 2.5\alpha(\alpha < 2)$, and $(R_\infty - R_0)/R_0 = 3.25\alpha^{0.705}(\alpha > 2)$.

2.3 Effect of Shearing

As a first step, we initialize a number of particles (parameters in Table 2.2), each with velocity (quite arbitrarily) 10 cm/s in random directions. They settle under gravity in a cavity that is static (Figure 2.3(a)) or has a moving bottom wall with velocity of 1 cm/s (Figure 2.3(b)). The interaction force among the particles and between the particles and

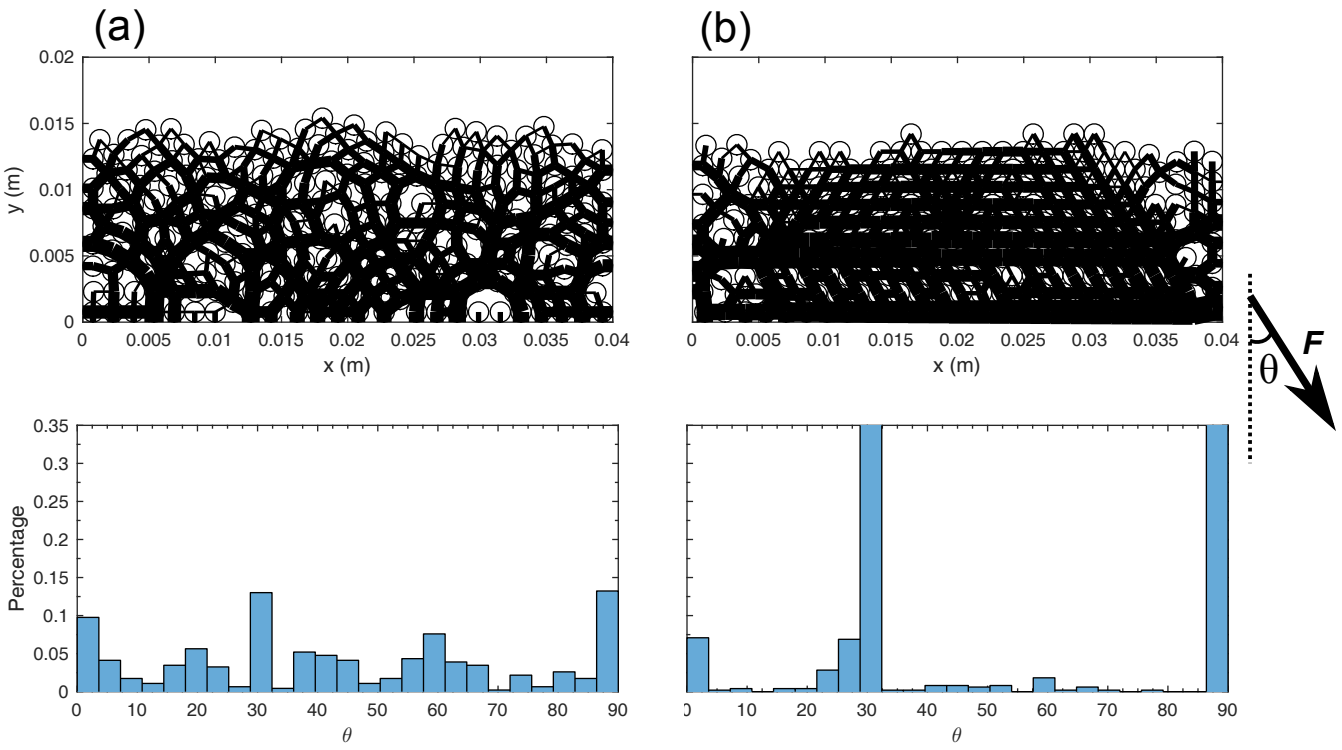

Figure 2.3: Visualization of particles and force network of a random deposit of particle in a static cavity (a) and a cavity with bottom wall velocity at 1 cm/s (b). The bottom part of each panel shows the distribution of the angle θ between each contact force and the vertical direction.

the walls are shown in Figure 2.3 for the two cases. Note that the discrete nature of the particles renders those interactions as a network, which is termed as a "force chain" or "force network" [Cates et al., 1998, Radjai et al., 1999]. The difference is quite apparent. The random nature of panel (a) is transformed into a more regular structure by the (weak) shearing, and the bottom layer of particles are all in contact as expected. The transition from a random packing to a more regular one is also visible in the histogram plot that shows the angle distribution of each normal contact force θ. The regular structure has spikes at angle equals 30 and 90 degrees, which corresponds to the arrangement of a triangular lattice.

2.4 Parametric Experiment

The parameters used for our simulation are listed in Table 2.2. The particle-wall interaction parameters (p-w values in the table) are applied to the *bottom* wall only, for the

Table 2.2: Parameters for jamming simulation

Parameter	Value
Particle diameter	1/16 in
Particle density	7833 kg/m3
Young's modulus	2×10^8
Normal restitution coeff. p-p, p-w	0.9, 0.9
Friction coefficient p-p, p-w	0.6, 0.8
Cavity width	4 cm
Time step	$1 \times 10^{-6} s$

two side walls the same p-p values are used. We initialize particles as a square lattice in a box region $[1cm, 3cm] \times [0, H]$, where H is a variable to control the amount of particles. The second parameter we varied is the belt speed U. Figure 2.4 shows the time series of the particles' averaged translational kinetic energy. Each panel is for a fixed H, and each curve in it corresponds to a certain belt velocity U. As expected, a system with large H and very small U won't be able to move at all, while one with small H and large U circulates continuously. But even in this ideal setup, the system alternates between the jamming and flowing states for certain combinations of (H, U).

2.5 Force Network Analysis

To analyze the transition, we choose the case ($H = 7\,cm$, $U = 0.7\,m/s$), a zoomed in time series of its averaged kinetic energy is shown in Fgiure 2.5. We look at the detailed interactions among the particles.

2.5.1 Flowing to Jamming Transition

An example of this transition is the box (a) of Figure 2.5. The particle positions and the force network are extracted and shown in Figure 2.6. It can be seen that in the circulating state the system are quite energetic, there are only few particle-particle contacts. The beginning of the jamming starts with the "freezing" of the bottom layer of particles. Particles in this layer can become stationary due to the hindering from the right wall. Since the system's energy source is the belt, the freezing of the bottom layer insulates the energy input to the system. At the same time, particle-particle collisions continue dissipating the

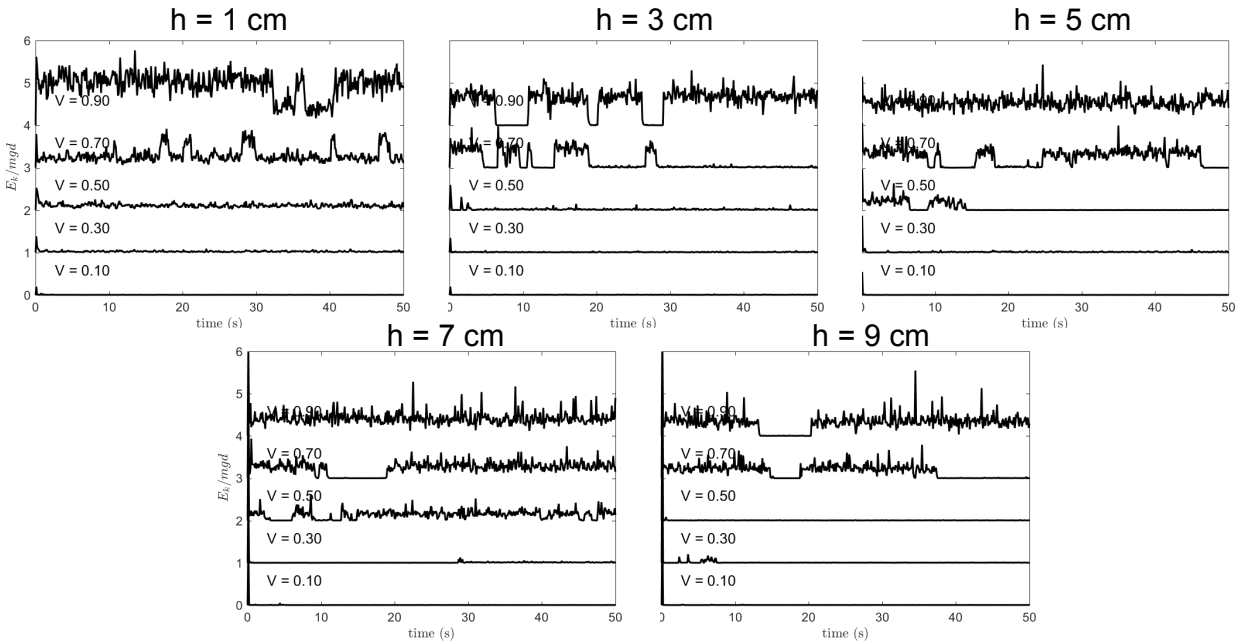

Figure 2.4: Time series of the particles' averaged translational kinetic energy. Each panel is for a fixed H, and each curve in it corresponds to a certain belt velocity U. The higher energy part of the curve corresponds to particle circulation or a "flowing" state, while the low energy corresponds to the "jamming" state.

existing energy. As a result of this energy loss, more internal particles are in continuous contact with each other, and a larger force network is formed. The growth of the network starts from the bottom left corner, where the particle motion stops first. In fact, the particle colored red never changed its position during this transition.

The system's jamming state is characterized by a very regular, hexagonal force network in the middle. The border of this regular region appear to take the shape of an equilateral triangle. Outside of this triangle, at the upper left and upper right regions, the force network is irregular structured.

2.5.2 Jamming to Flowing Transition

Figure 2.7 shows snapshots of the transition of a jamming state to flowing state (box (b) of Figure 2.5). The system in the jamming state is not really stationary, as shown by the constant loss of contact of particles close to the free surface. The reason is the shearing of the belt still inputs energy to the system through tangential friction that rotates the

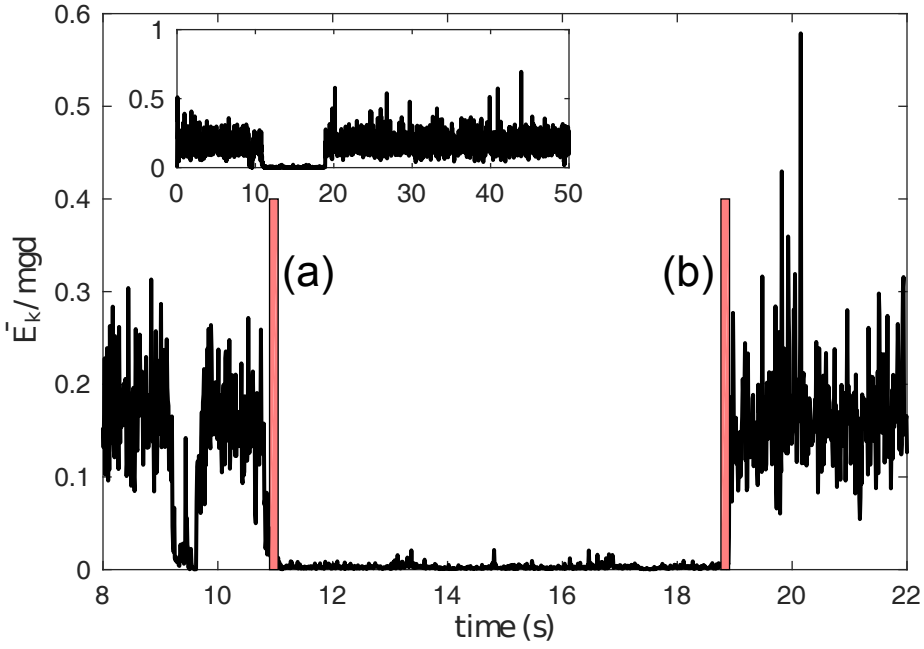

Figure 2.5: Time series of the averaged (over all the particles) translational kinetic energy, the parameters are $U = 0.7m/s$ and $H = 7cm$. The energy is nondimensionalized by mgd, where m is particle mass, g is gravity constant, and d is particle diameter. The figure is a zoom-in of the entire time series of $50s$ (inset). Three colored boxes ((a) and (b)) mark the time ranges to be analyzed in detail in the next section.

bottom layer particles. The motion transfers to upper particles also through the tangential forcing between particles. Even thought the overall structure of the force network does not appear change much, the slow evolution of the system could be noticed by, for example, following the particle colored red. Figure 2.8 shows its y position in each snapshot.

The mechanism of this transition can be discussed more clearly using Figure 2.1. Assuming a jammed state void of any motion, as the belt starts moving from below, the shearing inputs energy into the system through layer 1 particles' rotational degree of freedom. If nonslip contacts are maintained, layer 1 particles have to rotate counter-clockwise, while layer 2 particles have to rotate clockwise, which is opposite to that of layer 1 particles. These rotations translate into translations in accordance with the nonslip constraint, thus layer 1/2 particles translate to the right/left. In other words, this "upward" energy propagation is accompanied by the alternation of rotation and translation direction, or shearing,

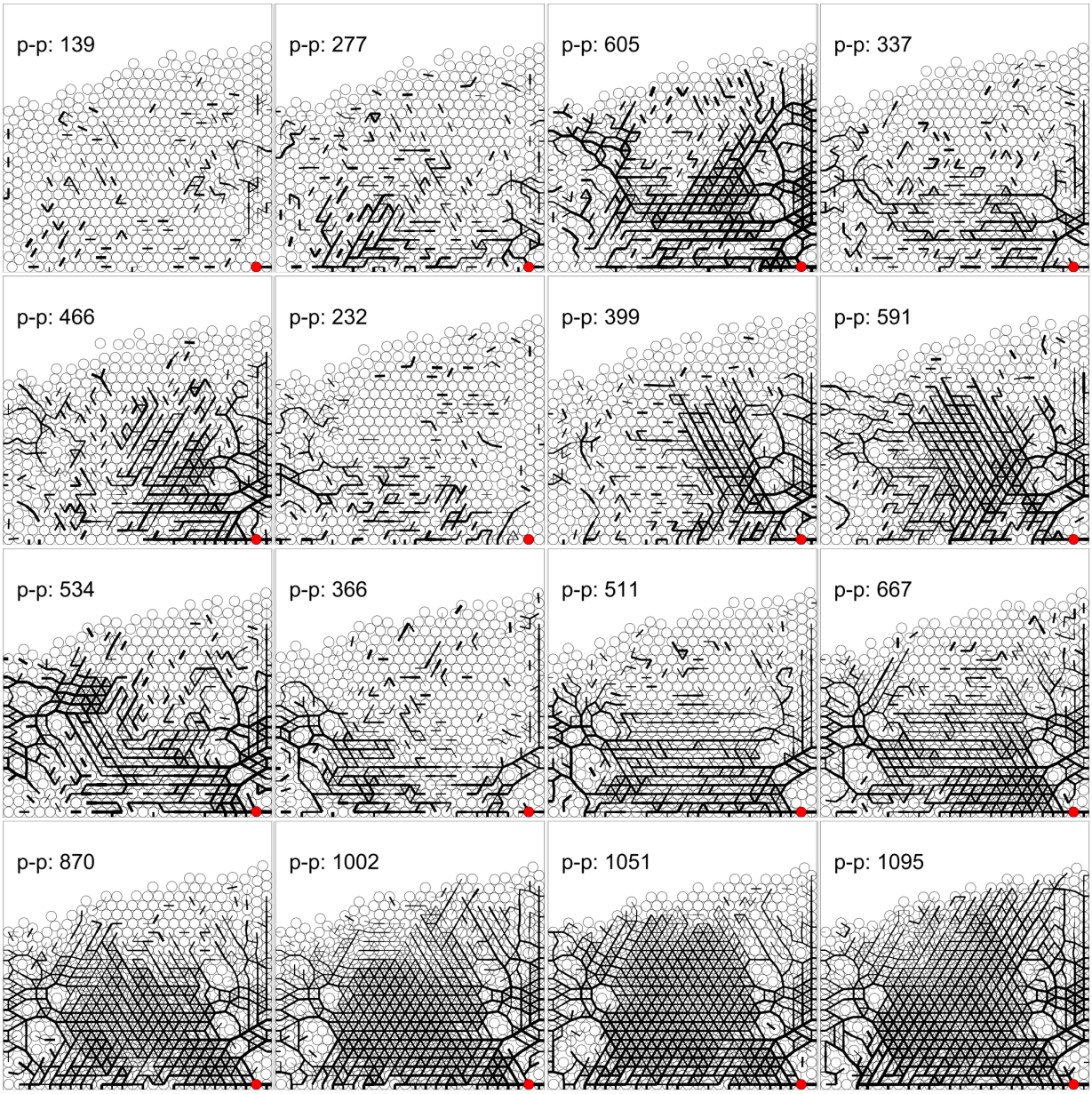

Figure 2.6: Visualization of particles and force network of the time range (a) of Figure 2.5. The text starting with "p-p" displays the number of particle-particle contacts in the system. The relation between the line width of the force network is determined from the force magnitude as $1 + \max(0, 1/2 \log_2(|F|/mg))$ One particle is colored red for tracking. The time of each snapshot increases by $0.01s$ towards the right and downward direction.

between layers. Of course, a particle may slip and violate the non-slip assumption. In fact, we simulated cases where both particle-particle and particle-wall friction coefficients are reduced, and found that particles remain static for the entire simulation time of $50s$.

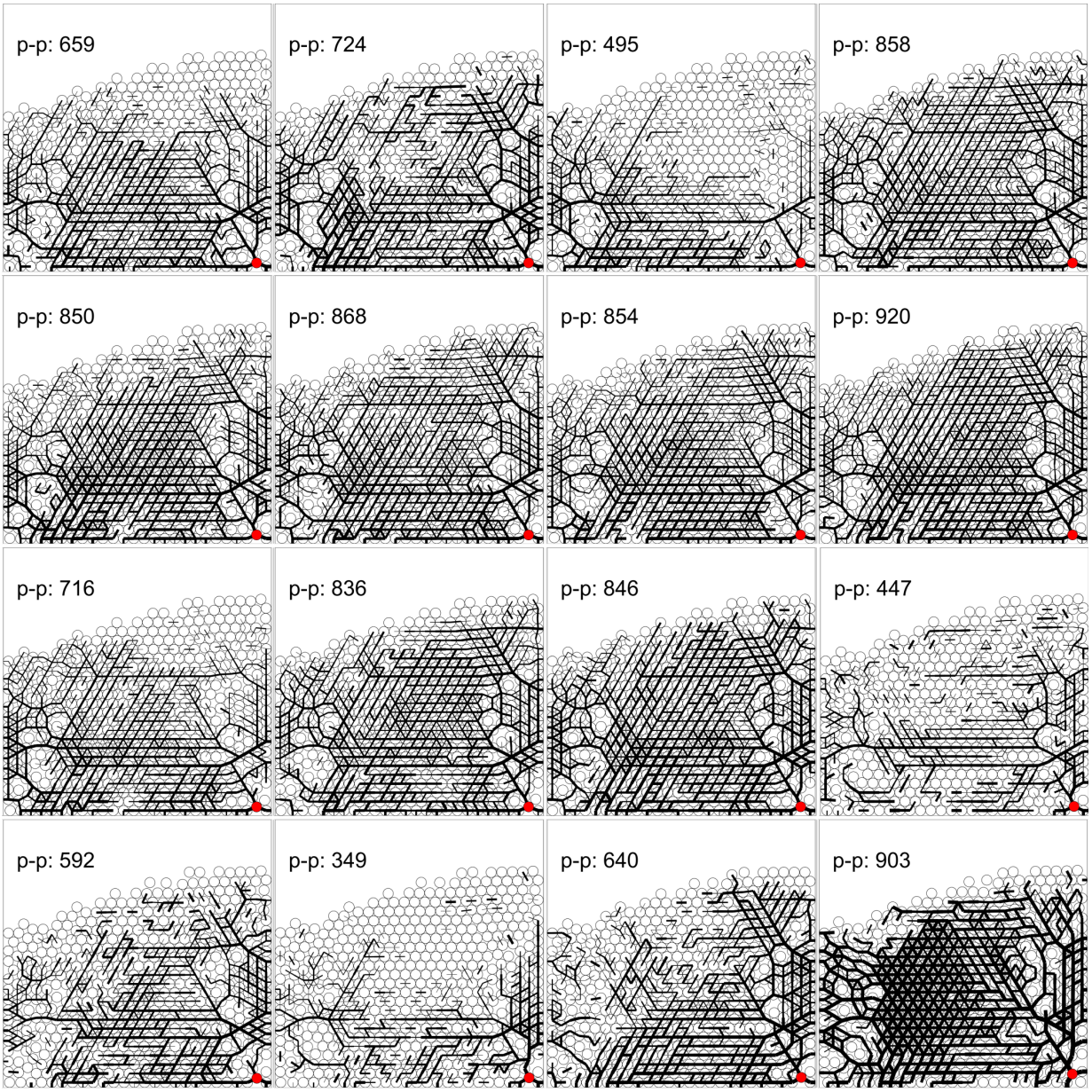

Figure 2.7: Similar to Figure 2.6, for the time range (b) of Figure 2.5.

2.6 Conclusion

This chapter presents a surprising phenomenon of the alternation of a 2D particle system between jamming and flowing states, which can be explained by the force network. The particle-particle friction (and the rotational degree of freedom) provides the key mechanism. A more quantitative study will be continued in the future.

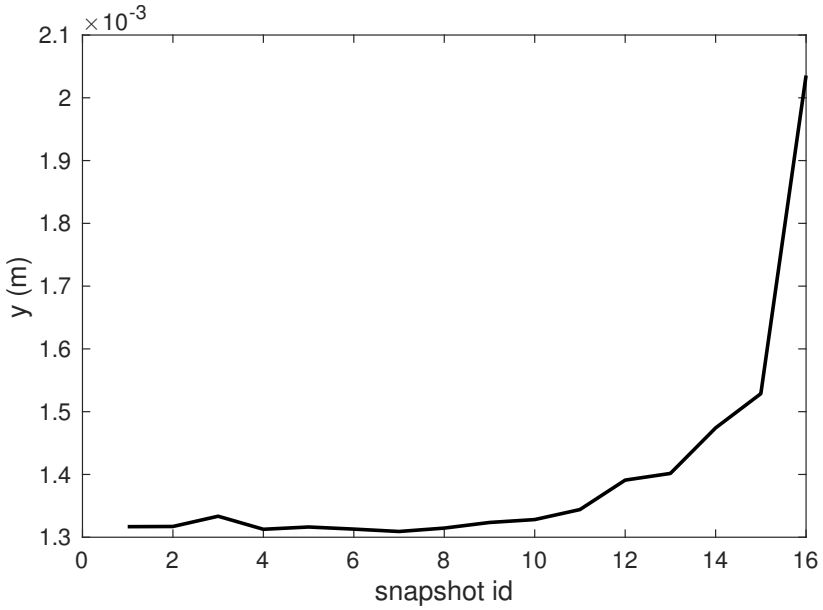

Figure 2.8: The y position of the red particle in Figure 2.7. The x axis is the index of the snapshots, starting from the upper left as number one and increase towards the right and bottom direction.

Chapter 3
Application of Signed Distance Field in Immersed Boundary Calculation

3.1 Introduction

Solid bodies in computational fluid dynamics (CFD) are usually represented as surfaces, which after meshing becomes (part of) the boundary of a fluid volume mesh. The generation of high quality mesh is often time consuming to resolve the complex geometry features from solid bodies. In addition, the mesh quality can be adversely affected if the solid body is moving, and may need smoothing or even re-generation on the fly. Many methods have been developed to reduce or even remove this complexity. Notable examples include mesh-free methods (like smoothed particle hydrodynamics [Monaghan, 2012]), Cartesian grid methods [Ye et al., 1999], and immersed boundary method [Peskin, 1977, Mittal and Iaccarino, 2005].

The immersed boundary method, or IBM, was proposed by Peskin [Peskin, 1977] in the 1970s to study the interaction between fluid and flexible structures. It has been extended by many researchers to simulate fluid solid interactions and has become an important simulation approach of particulate flows [Uhlmann, 2005, Fogelson and Peskin, 1988]. In IBM, the fluid mesh is usually Cartesian. The solid boundaries, instead of being meshed explicitly, directly immerse in the fluid domain. The effect of solid boundary on fluid flow, or the no-slip boundary condition, is imposed by adding certain forcing terms to the Navier-Stokes equations. Two variants, named continuous forcing and discrete forcing are common practices in immersed boundary method [Mittal and Iaccarino, 2005]. The former has force terms in the continuous N-S equations before discretization while the latter applies forcing terms after numerical discretization. In simulations of rigid bodies in fluid, the discrete forcing method is preferred, as the continuous forcing method often results in a stiff system that suffers from severe numerical instabilities [Lai and Peskin, 2000].

Interpolation is widely used method in many studies to impose the non-slip boundary condition in the discrete forcing formulation of immersed boundary method [Ye et al., 1999, Peskin, 1977, Tseng and Ferziger, 2003]. The idea is to reconstruct the fluid velocity at mesh nodes close to the solid boundary, such that the no-slip boundary conditions on the solid boundary are satisfied. Details can be found in the referred work [Ye et al., 1999, Peskin, 1977, Tseng and Ferziger, 2003]. Another approach, which the present work is based upon, uses volume forcing [Kajishima and Takiguchi, 2002, Yuki et al., 2007] to approximately satisfy the no-slip boundary condition. This method handles the fluid-solid interaction (and hence the no-slip boundary condition) as an extra step after each time stepping of the Navier-Stokes equations which are discretized in a finite volume form. In this step the fluid velocity of the cells fully contained by the solid body is set to be the solid velocity. At cells that are partially contained, i.e., intersected by the solid boundary, a special force is applied such that the cell velocity would represent a solid volume fraction weighted average of the solid velocity and the fluid velocity (more details in Section 2.3). This method has shown good accuracy in immersed boundary simulations and its numerical implementation is straightforward [Kajishima and Takiguchi, 2002]. However, the accuracy of this approach relies heavily on the accurate calculation of the solid volume fraction, which is used as a weighting factor of the forcing terms to transit smoothly the solution from solid to fluid.

Various methods have been proposed to calculate the solid volume fraction—hereafter represented by ϵ. The simplest method [Liao et al., 2010] is to set $\epsilon = 1$ for cells whose center falls within the solid, and 0 otherwise. Kajishima et al used locally reconstruct tangential plane at each cell to calculate the solid volume fraction of spherical particles [Kajishima and Takiguchi, 2002]. Yuki et el [Yuki et al., 2007] designed an empirical formula for solid volume fraction calculation, which was adapted in several other immersed boundary implementations[Bigot et al., 2014, Gsell et al., 2016]. Solid volume fraction is also used in methods other than immersed boundary. For example, in simulating particulate flows Sharma and Patankar [Sharma and Patankar, 2005] used solid volume fraction in their

Lagrangian multiplier based formulation. They approximate it by subdividing a fluid cell and counting the number of the sub-cells that fall inside the solid; the percentage of the inside sub-cells provides the estimation of the solid volume fraction.

In this work we show that a signed distance field representing the solid bodies can give second order accuracy in calculation of the solid volume fraction ϵ. We also showcase the capability of signed distance field applied to immersed boundary method for fluid structure interaction problems. Since the implementation is based upon OpenFOAM [Jasak, 1996], we give a brief introduction to OpenFOAM in the next section. Then we introduce the immersed boundary algorithm. The concept of signed distance, and its application in 2D and 3D volume fraction calculations and discussed in detail next. We present three volume fraction calculations and three immersed boundary calculations to validate our method and implementation. Concluding observations are made in the last section.

3.2 Introducing OpenFOAM

OpnFOAM stands for Open Field Operation & Manipulation. The "field" refers variables that are continuous in space and modeled with partial differential equations. Example application domains include fluid dynamics, heat/mass transfer, stress analysis.

OpenFOAM is written in C++ and make extensive use of advanced language features and software engineering concepts. For example, the object orientated design makes it possible to write partial differential equation solvers in a form very similar to the original mathematical expression. The main advantage of OpenFOAM is its well designed levels of abstraction and modules, making it easy for the user to plug in their implementations with minimal modification of the original code base. OpenFOAM is open source and released under GPL license, so it is free to use and modify for both academic and industrial purposes. Apart from the software design, OpenFOAM is distributed with with many solvers and utilities. For example, there are utilities for calculating vorticity, sampling the field, integrating force and torque etc. All those features make it ideal as an infrastructure to develop computational algorithms for continuum mechanics research in general.

For our purpose, we are most interested in the CFD of incompressible flows. Basically OpenFOAM uses collocated finite volume discretization for transport equations, together with a face based flux field for mass balance. This flux field creates a pseudo staggered arrangement of velocity and pressure which avoids pressure velocity decoupling [Ye et al., 1999, Jasak, 1996]. It can be made second order accuracy in space and time. Other useful features for CFD include dynamic mesh, adaptive mesh refinement, domain decomposition and parallel execution, and Lagrangian particle tracking.

3.3 Immersed Boundary Scheme

The immersed boundary scheme itself is based on that from Kajishima and Takiguchi [2002]. The fluid solver uses the fractional step method and is detailed below step by step (also refer to Figure 3.1).

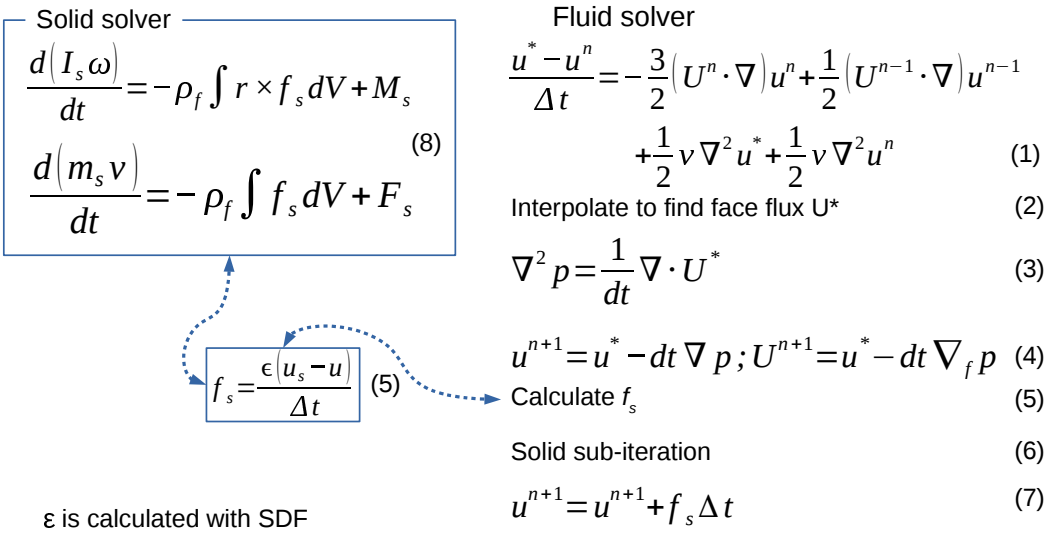

Figure 3.1: Summary of the immersed boundary algorithm.

(1) Predict a temporary velocity field u^* based on advection and diffusion of momentum. The pressure effect is excluded. We use Euler implicit scheme for temporal term, Adams-Bashforth scheme for the advection term, and Crank-Nicholson scheme for the diffusion term.

(2-3) u^* is not divergence free and needs correction. It is interpolated into cell faces to generate a flux field U^*. The divergence of U^* is used to solve for the pressure field that is

subject to incompressibility constraint.

(4) Correct the velocity field with the pressure calculate in last step. Cell center velocity is corrected by the pressure gradient evaluated at the cell center, while the flux field U^* is corrected by the pressure gradient at the cell face.

(5) Calculate the fluid-solid interaction force $\boldsymbol{f_s}$. This force is weighted by the solid volume fraction so is in effect only for fluid cells that are (partially) covered by the solid. This is where the signed distance field is applied to calculate ϵ (Section 2.4). Note that $\boldsymbol{f_s}$ is a discrete formulation as the concept of volume fraction only exists after discretization.

(6) A solid sub-iteration loop is used to calculate solid-fluid as well as solid-solid interaction. This sub-iteration with smaller time step is necessary in case of contact resolutions. Since in this chapter we only present cases with single solid, the number of sub-iteration is set to one. The solid's kinematic properties (position, orientation, velocity and angular velocity) are updated according to the rigid body dynamical equations of the solid.

(7) Update the fluid velocity using the same interaction force calculated in step (5). This ensures the satisfaction of Newton's third law.

It can be seen that the solid volume fraction ϵ has a direct impact on the interaction forces between the solid and the fluid. In the next section, we will introduce signed distance and demonstrate how it can be applied to find the solid volume fraction accurately.

3.4 Signed Distance Field

The signed distance field φ is a scalar field generated by a closed surface S such that, for any position $\boldsymbol{p}$ in space, its field value is the smallest distance from $\boldsymbol{p}$ to S. The common sign convention is $\boldsymbol{p}$ outside of the geometry has positive signed distance. Figure 3.2 shows the signed distance fields φ generated for three 2D primitive or elementary geometries (A-C) and for one after a Boolean union operation (D). The distinguished feature of φ is that its contours have very similar shape as the geometry. They appear like dilation or shrinkage of the geometry itself, which is the $\varphi = 0$ contour. The advantage of φ originates from the fact that it encapsulates the distance-to-geometry of the entire space. This simplifies

many geometric computations. For example, it is very convenient to combine geometries through Boolean operations. Suppose φ_1, φ_2 are the signed distance field of two geometries, then $\min(\varphi_1, \varphi_2)$ gives that of the union of two geometries (Figure 2D); $\max(\varphi_1, \varphi_2)$ gives that of the intersection of them; and $\max(\varphi_1, -\varphi_2)$ gives the signed distance of geometry one subtracted by geometry two. In this way complicated geometries are created from elementary ones like spheres or boxes. Many more features and extensive use of signed distance can be found in the field of computer graphics, examples include image processing and rendering [Jain, 1989, Green, 2007], collision detection [Teschner et al., 2005], and so on.

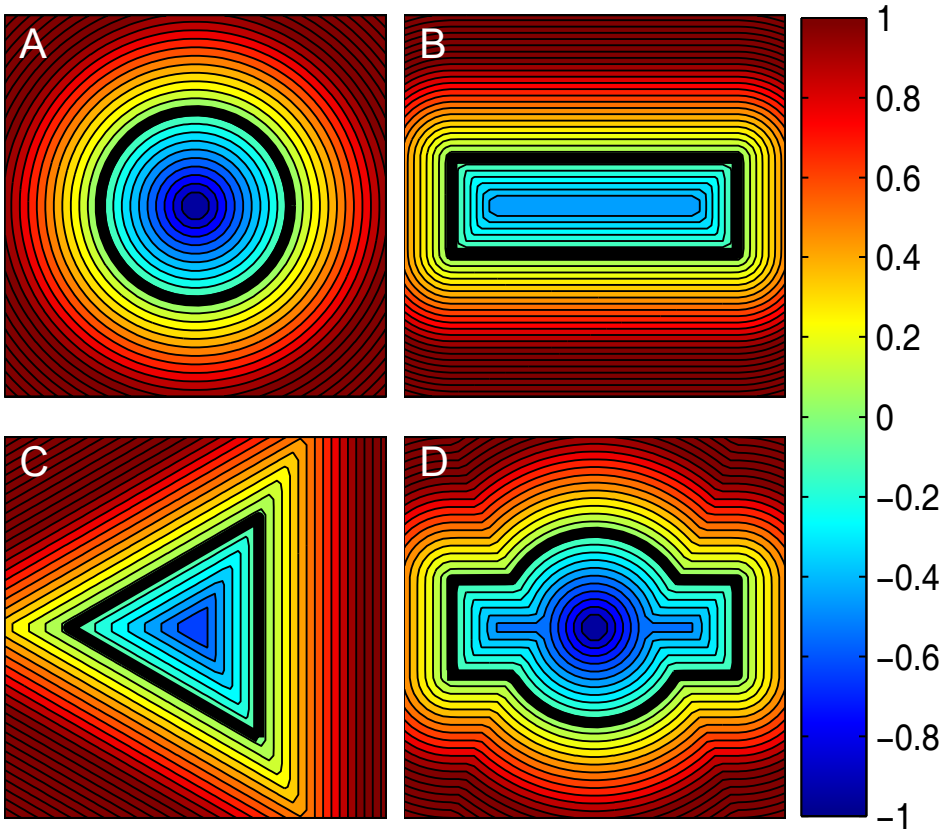

Figure 3.2: Example signed distance field φ of different shapes in $[-2, 2] \times [2, 2]$. A: a unit circle B: a 3×1 rectangle centering at origin; C: an equilateral triangle with height 2; D: a shape formed by the union of the unit circle in A and the rectangle in B. The color range is clipped to $[-1, 1]$. The convention is the field is negative inside the shape and positive outside of the shape.

A Cartesian mesh is used to discretize the N-S equations in our immersed boundary method. To use signed distance field to calculate solid volume fraction ϵ, we assume all finite volume cells that the solid boundary possibly intersects are squares in 2D or cubes in 3D. They are simpler to demonstrate and do not impose much restriction in applications. In addition, the algorithm presented for square/cube cells are readily extended to rectangle/box cells with minor modifications.

Given any geometry with solid boundary represented by $f(\boldsymbol{r}) = 0$ and any control volume or CFD cell, we first find if the cell is completely inside or outside of the geometry. This can be done by substituting each of the cell vertex $\boldsymbol{v}$ into f and check the sign of $f(\boldsymbol{v})$. If they all have positive sign, then (by our convention) the cell is completely outside of the geometry and has $\epsilon = 0$. If they all have negative sign, then the cell is contained with the geometry and has $\epsilon = 1$. The more complex scenario is when the cell is intersected by the geometry boundary and has fractional value of ϵ. This is exactly where SDF is used for accurate evaluation. We assume that all geometry features are linearly represented at the CFD cell level considering fine mesh is always necessary to resolve the geometry features and thus capture the flow details near solid boundary. With this assumption the geometry-cell intersection reduces to plane-square intersection in 2D, and plane-cube intersection in 3D, which are discussed below separately.

3.5 Two Dimensions

The number of a cell's vertices v_i that satisfies $f(v_i) < 0$ classifies all intersections into three basic cases (Figure 3.3), other cases are just rotational duplicates. Take case A in Figure 3.3 as an example, φ_a and φ_d gives the distances from vertex a and d to the plane, respectively. The distances are marked in the figure with dashed lines. Since by definition φ_a and φ_d are the shortest point-to-surface distances, the dashed lines are perpendicular to the plane. By similarity of triangles we find the edge fraction $\epsilon_2 = \varphi_a/(\varphi_a + \varphi_d)$. The other nonzero edge fraction, ϵ_1, follows similarly. To avoid division by zero in singular cases (such as plane passing a vertex) we set $\varphi = -10^{-9}$ if $|\varphi| < 10^{-9}$. This small offset has negligible

effects on the physical simulation but helps the robustness of geometric calculations. After getting the edge fractions, face fractions of all three cases in Figure 3.3 follow directly from elementary geometry:

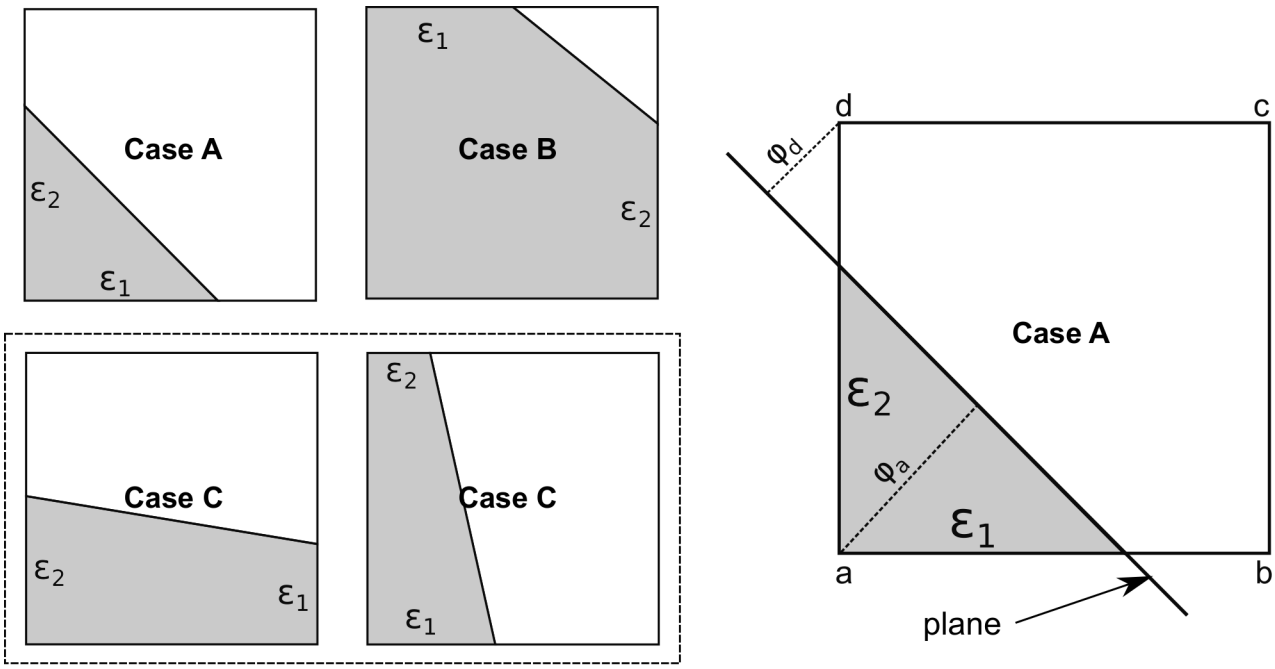

Figure 3.3: The left subplot shows the cases of plane-square intersections when the square's bottom left vertex is within the geometry. The bottom two cases are similar and both considered as case C. Totally there are three basic intersection cases. The plane is a local approximation of a solid geometry (shaded). The right subplot is an enlarged view of case A, with detailed annotations for discussion in the text.

- Case A: from triangle area formula $\epsilon = (\epsilon_1 \cdot \epsilon_2)/2$.
- Case B: one minus the triangle area $\epsilon = 1 - (1 - \epsilon_1) \cdot (1 - \epsilon_2)/2$.
- Case C: from trapezoid area formula $\epsilon = (\epsilon_1 + \epsilon_2)/2$.

Note that ϵ is dimensionless, so both multiplication and addition between ϵ_1 and ϵ_2 are valid operations.

3.6 Three Dimensions

Unlike two dimension intersections where we can enumerate all possible scenarios, three dimension intersections are more complicated. The intersection face can have number of

sides range from three to six depending on the plane location and orientation. For example, the marching cube algorithm (for generating contour surfaces on a cubic grid) [Lorensen and Cline, 1987] enumerate all the 256 intersection cases. Although the cases can be reduced to fewer number of basic cases by rotation and reflection symmetry (as done in [Lorensen and Cline, 1987]), and some ill-posed cases will not appear in our application, we prefer a direct approach to avoid branching conditions in code, and avoid the many analytical expressions needed for each basic case.

The problem is essentially to find the volume of the polyhedron formed from the rest of a cube cut by a plane. Our method is based upon the fact that this volume can be computed by choosing an arbitrary point $\boldsymbol{p}$ in space (could be outside of the polyhedron), and summing volumes of each of the pyramids formed by every polyhedron face and $\boldsymbol{p}$. The only requirement is the normal of each face points towards inside of the polyhedron. Instead of first building the polyhedron and decomposing it, however, we directly decompose the cube itself. Figure 3.4 illustrates the idea applied to two dimensions. A plane cuts a square from the bottom left. The solid part is the colored triangle (shown as two parts in different colors). The square is decomposed into four triangles, each shaded with different line pattern. Suppose the square size is δ, take triangle *pab* as an example, its area is $1/2\delta\epsilon_1 h_1$. Note that edge fraction ϵ is used to weight the triangle's "bottom edge" so that only the purple shaded part of Pab contribute to the solid areaas it should. This is similar for triangle *pda* which also makes partial (colored in orange) contribution. The other two triangles *pbc* and *pcd* have zero bottom edge fractions and thus zero area. The reason that we can decompose the entire square to calculate the solid areainstead of decomposing just the solid partis because edge fraction weights automatically remove those triangles/pyramids that do not contribute. This is simple to implement and avoids many branching conditions.

The above method, which decomposes square into four triangles (or decompose a cube into six pyramids in three dimension), only works if $\boldsymbol{p}$ located exactly on the plane; otherwise we need to deal with the pyramid formed by the intersection face (which can have

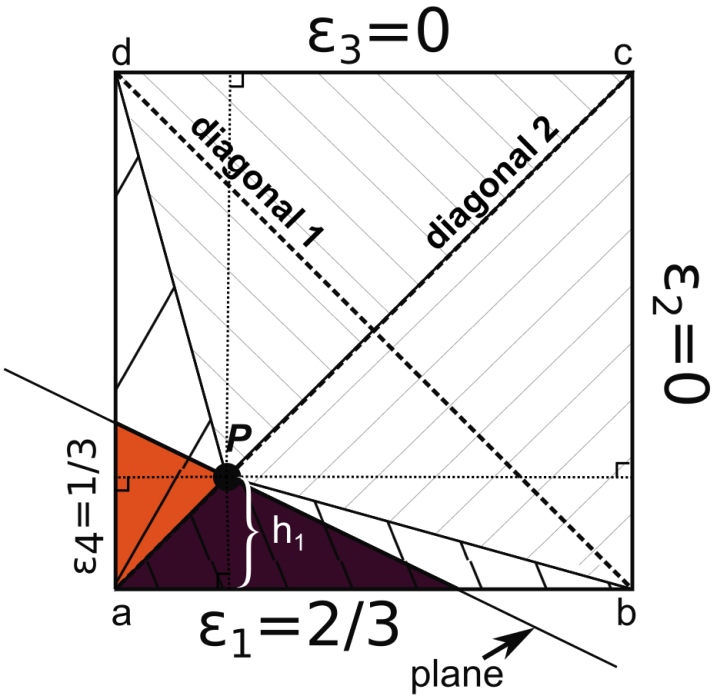

Figure 3.4: Two-dimensional illustration of the decomposition method for volume fraction calculation, and calculation of point p on the plane.

three to six sides) and $\boldsymbol{p}$, which depends on specific plane location and orientation and is much complicated. By choosing $\boldsymbol{p}$ on the plane, this pyramid has zero height and volume, and can be safely neglected in computation.

The problem with finding $\boldsymbol{p}$ is we don't have an explicit plane equation/definition-only eight signed distance values at the cube vertices are available. On the other hand, any point with zero signed distance can be used as $\boldsymbol{p}$. One way to find such as a point is to first build $\nabla\varphi$ from the eight φ values using central difference, then move an arbitrarily selected cube vertex along the gradient direction until it attains zero φ value. Another way, which is used by us, is based on the geometry intuition that if a cube is intersected by a plane, at least one of its four diagonals crosses the plane-this crossing point is then used as point $\boldsymbol{p}$. We illustrate this idea with the same two-dimensional case in Figure 3.4. Of the two diagonals, the second one crosses the plane, which is found by checking every diagonal pair of the square vertices (two pairs here: $a-c$ and $b-d$). With φ_a and φ_c having opposite

signs, point P is found by linear interpolation:

$$w = \frac{|\phi_a|}{|\phi_a| + |\phi_c|}, \boldsymbol{p} = (1-w)\boldsymbol{a} + w\boldsymbol{b}$$

The discussion using the two-dimensional examples applies to the three-dimensional plane-cube intersection, with the only difference being there are four diagonals to check to find $\boldsymbol{p}$, and six pyramids to sum for the polyhedron volume. The bottom face area of each pyramid is the cube face area δ^2 weighted by its solid area fraction ϵ_i, i being the face index from 1 to 6. ϵ_i is calculated using the two-dimensional method from the previous subsection. The pyramid height (see the dotted lines in Figure 4) is calculated from the face normal $\boldsymbol{n}_i$ and face center f_i by dot product $\boldsymbol{n}_i \cdot (\boldsymbol{p} - \boldsymbol{f}_i)$. At the end, we have a volume fraction expression:

$$\frac{1}{3\delta} \sum_{i=1}^{6} \epsilon_i \cdot \boldsymbol{n}_i \cdot (\boldsymbol{p} - \boldsymbol{f}_i)$$

The two algorithms (2D and 3D) presented above is accurate if the geometry feature is linear, which means they are second order accurate.

3.7 Approximate Signed Distance

The signed distance φ for certain elementary geometries, like a circle or sphere, has simple explicit expressions. For example, the unit circle in Figure 3.3A has $\varphi(p) = |p| - 1$, $|p|$ is the radial distance of $\boldsymbol{p}$. For most geometries however, φ is not readily available. Take the standard ellipse $x^2/a^2 + y^2/b^2 = 1$ as an example. Given any point $\boldsymbol{p}$ in space, we need to find the smallest distance between $\boldsymbol{p}$ and the ellipse. With the standard parameterization, the problem is to minimize $|\boldsymbol{p} - (a\cos(t), b\sin(t))|$. This requires solving nonlinear algebraic equations and is impractical considering the number of mesh cells and time steps in common immersed boundary applications. For this reason an explicit, approximate formula is derived. It is 2nd order accurate in space and can be used with the volume fraction algorithms in the last section without degrading its accuracy.

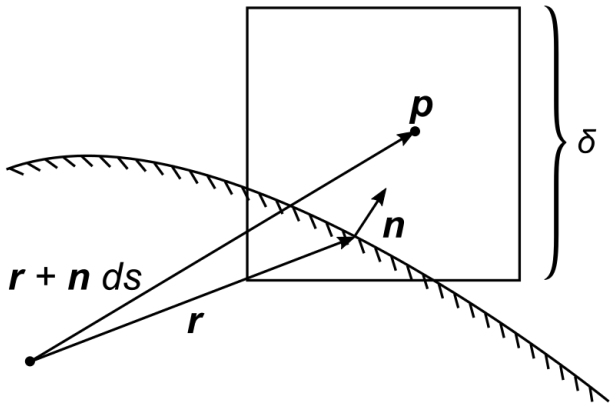

Figure 3.5: Illustration for the derivation of approximate φ of arbitrary geometry $f(r) = 0$. δ is the size of a fictitious mesh cell.

It is firstly noted that, in the calculation of the volume fraction field, φ is applied only to mesh cells directly intersected by the geometry. This means only the φ value within a thin shell around the geometry is needed, the thickness of this thin shell has the same order as the mesh cell size δ. Referring to Figure 3.5, suppose a point $\boldsymbol{r}$ on the geometry is displaced by distance ds along the normal direction $\boldsymbol{n}$, ds being the signed distance. This generates a new point $\boldsymbol{p} = \boldsymbol{r} + \boldsymbol{n}ds$. $\boldsymbol{p}$ does not satisfy the geometry equation $f(\boldsymbol{r}) = 0$. Let $f(\boldsymbol{p}) = E$ (for "excess"), Taylor expansion gives $E = \nabla f(\boldsymbol{r}) \cdot \boldsymbol{n}ds + O(ds^2)$, where $f(\boldsymbol{r}) = 0$ is used. The gradient $\nabla f(\boldsymbol{r})$ can be approximated by the gradient at point p with Taylor expansion: $\nabla f(\boldsymbol{r}) = \nabla f(\boldsymbol{p})\nabla\nabla f(\boldsymbol{p}) \cdot \boldsymbol{n}ds + O(ds^2)$. Replace $\nabla f(\boldsymbol{r})$ with $\nabla f(\boldsymbol{p})$ in the expression of E, we have $E = \nabla f(\boldsymbol{p}) \cdot \boldsymbol{n}ds + O(ds^2)$. Finally, substitute $n \approx \nabla f(\boldsymbol{p})/|\nabla f(\boldsymbol{p})|$ gives the approximate signed distance:

$$\varphi(\boldsymbol{p}) = \frac{f(\boldsymbol{p})}{|\nabla f|} + O(\delta^2)$$

where the cell size δ is used as a scale for ds. As a simple test (numerical experiments will be shown in the next section), we apply it to the unit circle in figure 2A. Consider a point $\boldsymbol{p} = (x, y)$, its exact signed distance is $\varphi = |p| - 1$. The approximation formula gives,

after some algebra manipulation, $\varphi = |\boldsymbol{p}| - 1 + [-(|\boldsymbol{p}| - 1)2/(2|\boldsymbol{p}|)]$: the square bracket contains the approximation error. Because we are only interested in point p within a thin shell around the circle, $|\boldsymbol{p}|$ has order 1, and $|\boldsymbol{p}| - 1$ has order δ. The error term then has order δ^2 as expected.

This finishes the method for calculating the volume fraction field. It is worth mentioning that with this field obtained, it is easy to derive other useful quantities such as the center of mass or moment of inertia. The next section will present the computational experiments we performed to test both the volume fraction algorithm, and its applications in the immersed boundary method.

3.8 Results

3.8.1 Geometric Calculation

Volumes of three *convex* geometries, a unit circle, a cosine function, and an ellipsoid, are used to test the signed distance based volume fraction algorithm. Because all deviations from the exact volume occur at the intersection cells, where geometries are assumed locally linear, the error in the total volume is accumulated from the error in each cell and can be used to evaluate the accuracy of our algorithm. This method requires the test geometries to be convex to avoid possible cancellations between negative and positive errors. For comparison, we calculate the total volume using the step-wise representation of the solid, i.e., assign $\epsilon = 1$ if a cell's center falls within the geometry, and 0 otherwisewe will refer to this method as step method, and our method as linear method due to the local linear approximation.

A. Unit Circle with Different Grid Resolutions The area of a unit circle inscribed in a square domain $[-1, 1] \times [-1, 1]$ is calculated with different mesh resolutions. The resolution varies from 1×1, 2×2, up to 20×20. Figure 6A summarizes the relative error for each resolution used. The step method has strong oscillations even with the finest mesh resolutions, while our linear method shows smooth second order convergence. The relative error for a 10×10 mesh is only 0.86%, and with the 20×20 mesh it is about 0.17%.

B. Cosine Bump with Different Amplitudes Another advantage of the linear method lies in the smooth variation of volume fraction as the geometry deforms/moves. Consider a bump feature defined by $y \leq A\cos(\pi x/2)$ in $[-1,1] \times [-1,1]$, the amplitude A is varied for parametric study. With the step method a change of A may either have no effect on the solid volume fraction if it is small, or cause many jumps in the field if it is large. Figure 3.7B illustrates this aliasing effect as by plotting the relative error of the bump volume calculated on a 20×20 grid against the bump amplitude A. This aliasing and discontinuous change in ϵ is problematic for both parametric studies and simulation of moving objects. To use the linear method, we apply the approximation formula derived previously to have:

$$\varphi_{app} = \frac{y - A\cos(kx)}{\sqrt{1 + A^2k^2\sin^2(kx)}}$$

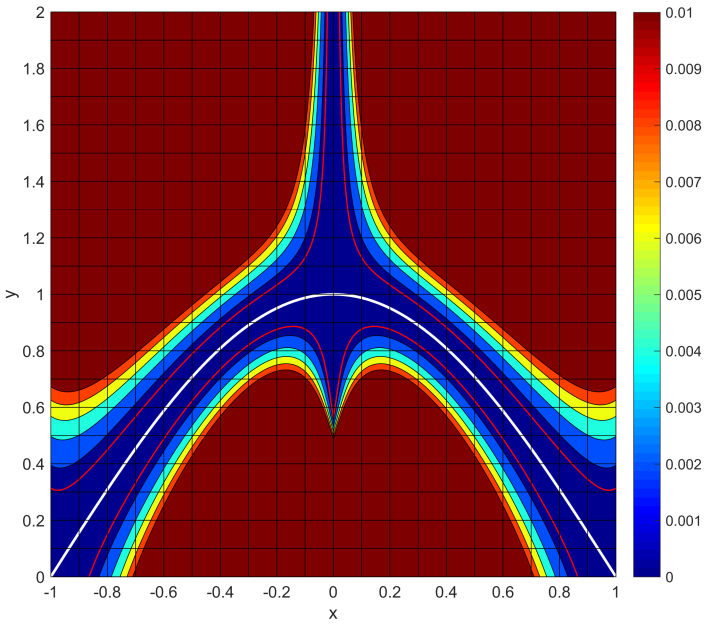

Figure 3.6: The absolution error $|\varphi - \varphi_{app}|$ of the approximate signed distance field φ_{app}. The reference field φ is calculated through an optimization routine. The white curve plots the function $y = \cos(\pi x/2)$. The two red curves are contours with value 0.001. The color range is clipped to [0, 0.01].

Figure 3.6 shows (absolute value of) the error in φ_{app} for the specific case of $A = 1$. The reference field is calculated numerically by solving t that minimizes $(y_0 - \cos(\pi t/2))^2 +$

$(x_0 - t)^2$, for any given location (x_0, y_0). The MATLAB function `fminbnd` is used, and t is considered converged when the relative change in its value is less than 10^{-4}. The error is indeed small around the curve. The overlaying 20×20 grid is also shown. For the intersected cells, the relative error of φ_{app} at their vertices is a few percent at most. Using φ_{app} and the algorithm in the previous section, the calculated bump volume agrees well with the exact value $2 + 4A/\pi$ and shows little sign of aliasing (Figure 3.7C).

C. Ellipsoid with Different Grid Resolutions A prolate ellipsoid (radius are $1/2, 1/2, 1$, exact volume $= \pi/3$) in box $[-1, 1] \times [-1, 1] \times [-1, 1]$ is used to test the three dimension method as well as the approximation formula in section 3.3. The smooth second order convergence is apparent in Figure 3.7C. The relatively large error for very small mesh resolution is likely caused by both the low mesh resolution and the approximate signed distance. For $n > 10$, the relative error is less than one percent. Results from the step method have strong oscillations even at fine mesh resolution, just like the unit circle case.

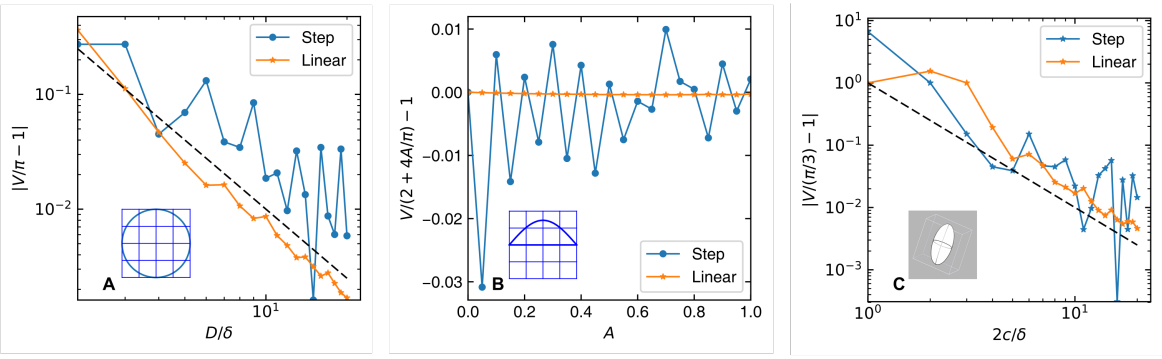

Figure 3.7: A: dependence of the relative error of circle area calculation on mesh resolution. The x-axis is mesh resolution in terms of the number of cells across the circle diameter ($D = 2$). Results from both step method and the linear method are shown. The dashed line shows the second order convergence. B: volume calculation of a cosine bump. The x-axis is the curve amplitude A. Results from both step method and the linear method are shown. C: volume calculation of an ellipsoid (radius are $1/2, 1/2, 1$) with different mesh resolutions: from 1×1 to 20×20. The dashed line represents the second order accuracy. In each panel the lower left inset illustrates the setup.

3.8.2 Spatial and Temporal Accuracy

The spatial accuracy of the immersed boundary method in Section 2 is estimated with a lid driven cavity setup, which has a unit circle obstacle at its center. The cavity domain spans $[-2, 2] \times [-2, 2]$. The circular obstacle obstructs and greatly changes the flow. We test a case where lid velocity equals 1, fluid viscosity equals 0.01. The Reynolds number equals 400 if the lid size is used as the characteristic length. Three Cartesian meshes with resolution 16×16, 32×32, and 64×64 are used to obtain the resolution dependent solutions, and a 400×400 body-fitting mesh is used to obtain the reference solution. The flow pattern of the reference solution is shown in Figure 3.8. The relatively complex flow pattern is a good test of our immersed boundary treatment. We use the u-velocity along the sample section in Figure 3.8 to evaluate the error in calculation. Because u-velocity of those cells intersected by the solid is a volume fraction weighted average of the fluid and solid velocity, we do not count them as solutions to the fluid equation and exclude them from error norm calculation. The error shows close to second order behavior (Figure 3.9), and suggests the volume average approximation at the solid boundary does not degrade the overall solution accuracy on the fluid side.

To test the temporal accuracy, we make two modifications to the previous cavity case. First, the top lid is made static; second, the circle is assigned a density value double that of the fluid, and is allowed to fall under gravity ($9.8m/s^2$). A fine 200×200 grid is used. Four time steps 0.001, 0.0005, 0.00025, and 0.0001 are used to obtain the resolution dependent solutions. A smaller time step of 0.00001 is used to obtain the reference solution. The temporal order is evaluated using the y location of the circle center at $t = 0.1$, and is shown in Figure 3.10. The temporal order of accuracy is lower than the expected second order from the scheme, similar observation was made in Bigot et al. [2014]. This is likely caused by the simple explicit coupling strategy of the fluid and the solid motion, as the immersed boundary forcing is applied after the velocity projection. For this reason there might be a temporal lag of Δt of between the fluid and solid dynamics. To improve the

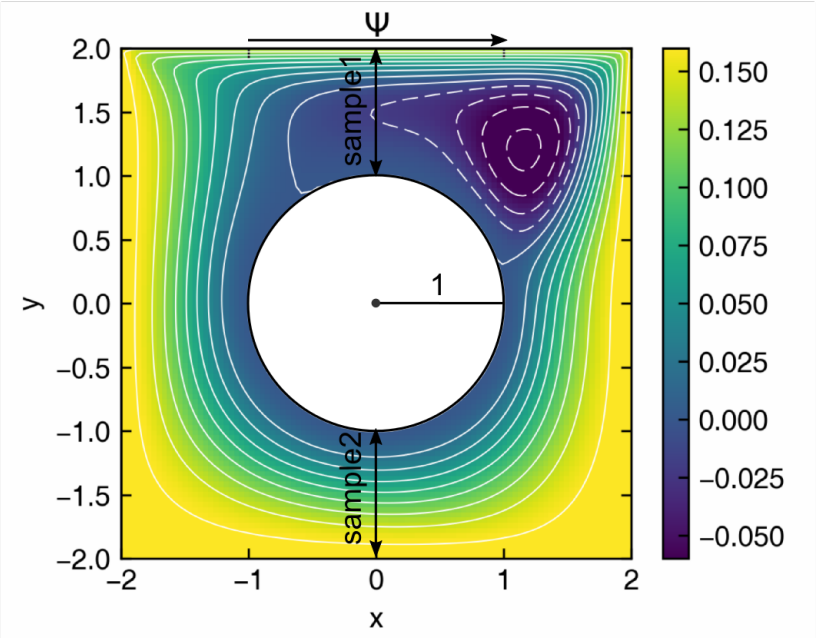

Figure 3.8: Lid driven cavity flow obstructed by a unit circle obstacle at $(0, 0)$, calculated by the reference solution. The stream function Ψ is shown in both shaded and contour plot. Ψ is offset by a constant such that it equals zero on the circular obstacle. The sample section used for error measurement is shown as well.

temporal order of accuracy an inner pressure correction iteration may help.

3.8.3 Free Falling Sphere in a 3D Box

The experiment by ten Cate et al. [2002] of a single sphere settling in a closed box container is simulated. The sphere hanging initially at $(50, 120, 50)$ in a $100 \times 100 \times 160$ box (all units are mm) is allowed to fall freely under gravity. Its vertical location and falling velocity are recorded as experimental data. There are four experimental setups using the same sphere, but different fluid properties (Table 1). The sphere diameter equals $15mm$, and its density equals 1120 kg/m3. A uniform Cartesian grid with $100 \times 100 \times 160$ cubic cells is used for simulation, the time step is $2.0 \times 10^{-4}s$. We use both step and linear methods for volume fraction calculation. Figure 3.11 plots the simulation results directly on the original figure inten Cate et al. [2002]. All other simulation parameters being the same, the differences in the results are caused by the volume fraction calculation method. We found using the step method the falling velocity and distance are systematically under-predicted.

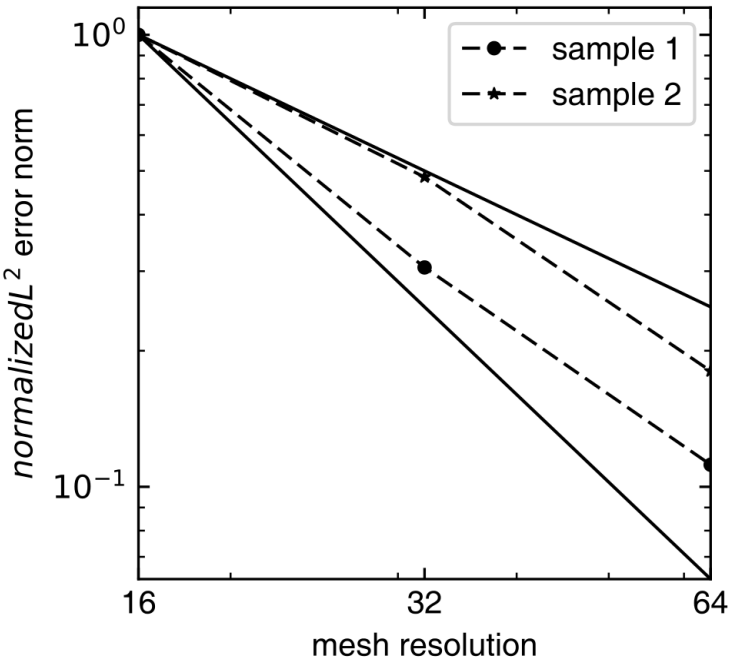

Figure 3.9: L^2 error norm versus mesh resolution. The errors are normalized by the first measured error on the 16×16 mesh. The solid lines represent the fist and second order convergences, respectively.

This is likely caused by the enhanced fluid drag caused by the effective zigzag surface representation. With the linear method the agreement between simulation and experiment is much better, and we recommend the use of accurate volume fraction calculations in future studies.

Table 3.1: Material properties used in experiment of free falling particle by ten Cate et al. [2002].

Case	$\rho(kg/m^2)$	$\mu_f(Pa \cdot s)$	Re
1	970	0.373	1.5
2	965	0.212	4.1
3	962	0.113	11.6
4	960	0.058	31.9

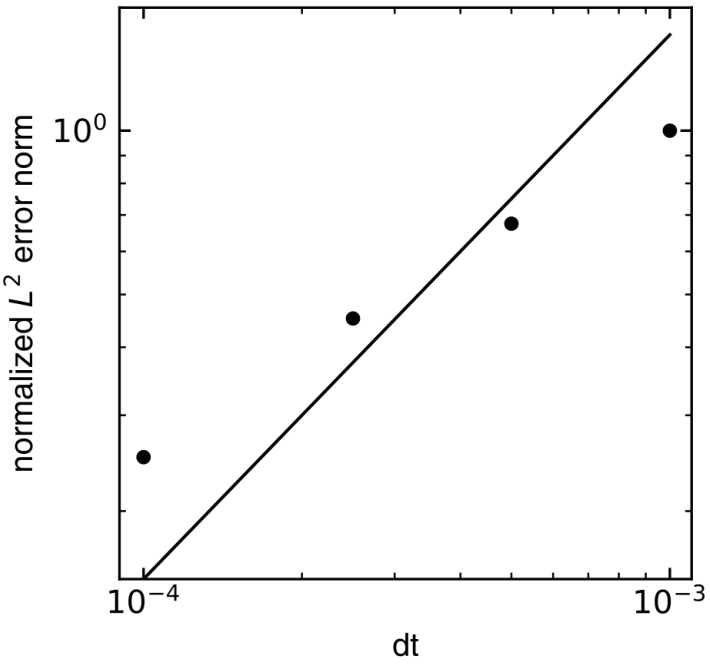

Figure 3.10: Temporal accuracy of the immersed boundary algorithm. The solid line shows the first order accuracy. The dots correspond to time resolutions of 0.001, 0.0005, 0.00025, and 0.0001. All errors are normalized by the error using time resolution 0.001.

3.8.4 Free Falling Cylinder with Appendage In 2D Box

Object with an appendage attached to it and falls under gravity was shown to develop the so called "invert pendulum instability" due to the interaction between the surrounding liquid and the appendage Lacis et al. [2014]. For certain Reynolds number and geometric parameters (mainly the appendage/object size ratio) the object reaches a steady state characterized by the appendage's tilt angle (annotated in Figure 3.12D), and the object's sideways drift angle α (annotated in Figure 3.12B). We use this case to test the Boolean method of forming complex geometries. The object is formed by the union of a cylinder (diameter equals 2) and a slender rectangle. The part of the rectangle outside of the cylinder has length equals 2, and width equals 0.3. Other parameters are object density equals 1.01, liquid density equals 1.0, liquid viscosity equals 0.01, and gravity constant equals 10. The

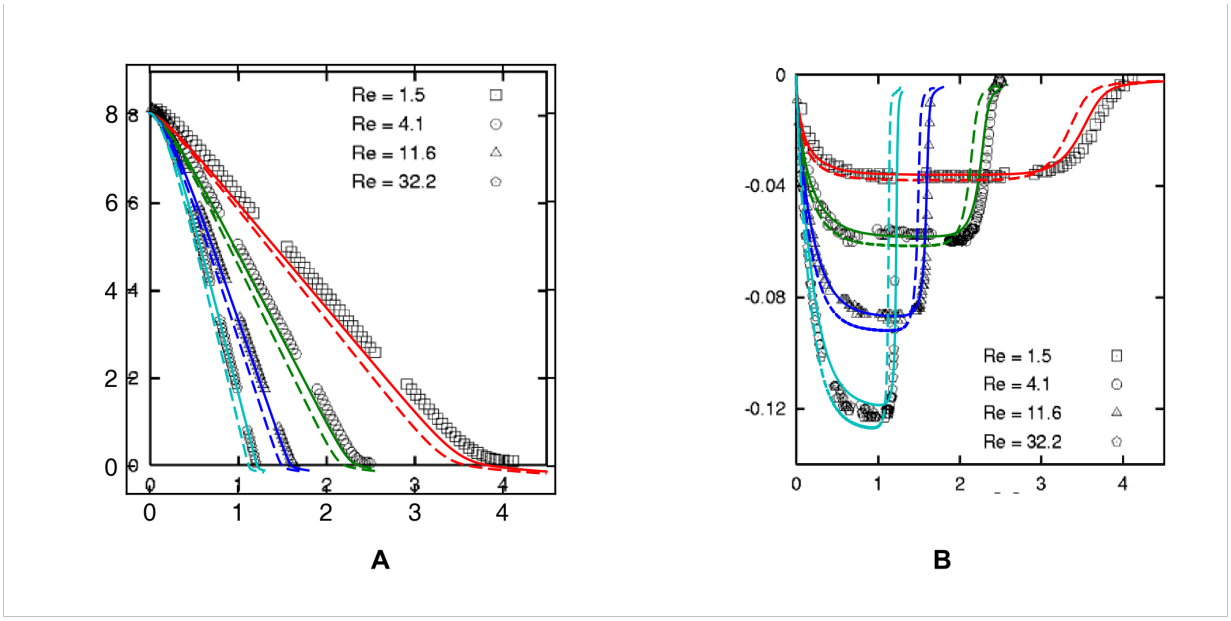

Figure 3.11: Comparison of numerical simulation using both step and linear volume fraction calculations against the experimental result: vertical position (A) and velocity (B). The simulations result are plotted directly on the original figure inten Cate et al. [2002]. The curves are simulation results using linear method (solid) and step method (dashed).

object initially locates at $(10, 190)$, and has an initial tilt angle equals 5 degrees to help the development of instability. It is released and allowed to fall freely under gravity.

Figure 3.12A shows a snapshot of the domain at $t = 340$. The flow pattern is visualized by line integral convolution technique to show the vortical structure. The object has already reached a dynamically steady state, the Reynolds number based on the terminal velocity, cylinder diameter, and liquid viscosity is about 100. Using the range of data during the dynamically steady state we can calculate the average drift and tilt angles, which are $\theta = 11.6$ degrees and $\alpha = 5.96$ degrees. One limitation we encountered was that due to the volume fraction based formulation, we were not unable to use as thin appendage as were used in the reference. The appendage is made thicker to have five cells across its width. Nonetheless, the essential physics are well captured by the immersed boundary method with our signed distance based geometry representation.

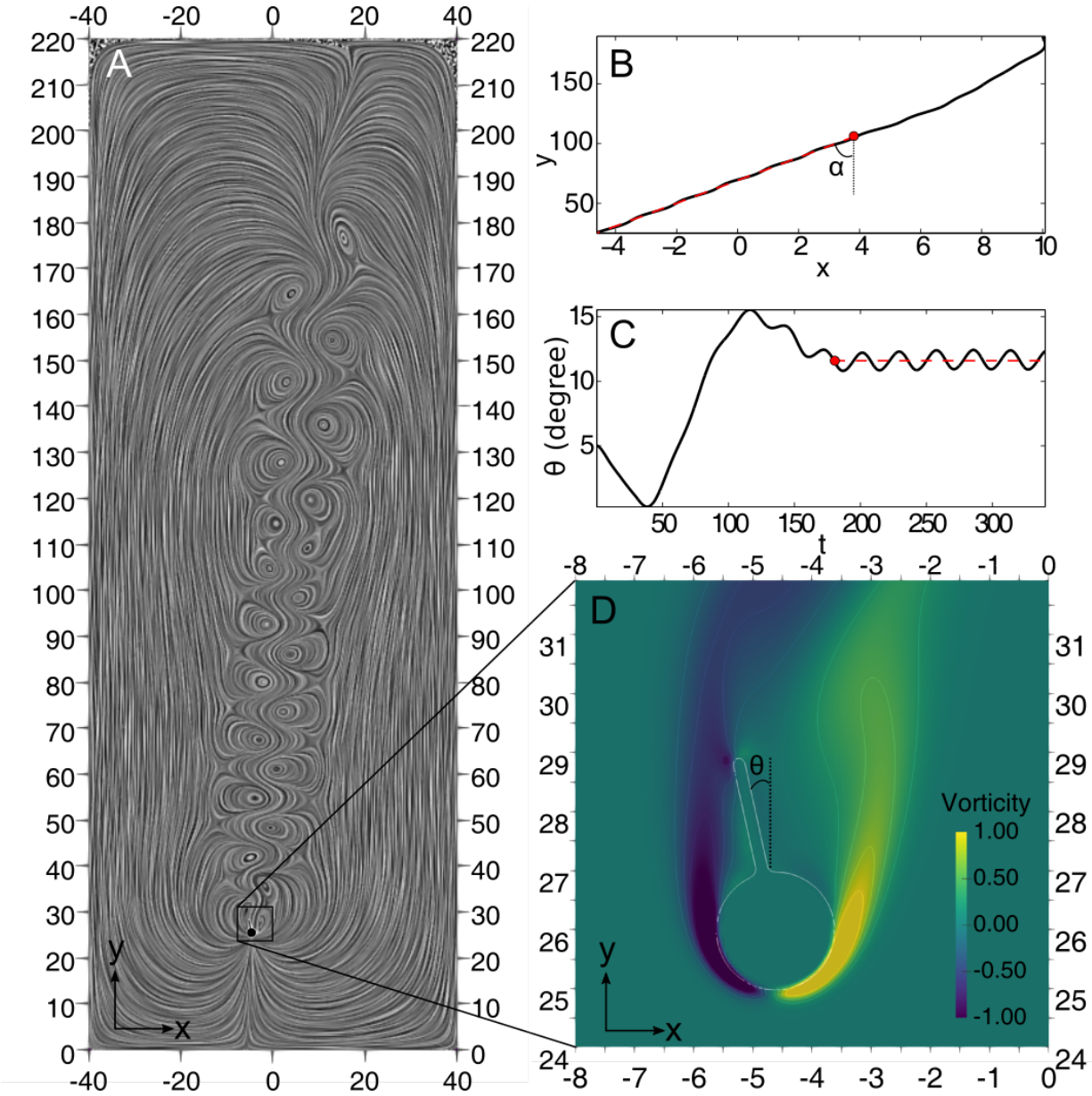

Figure 3.12: A: the vortical structure of the fluid at $t = 340$, visualized using line convolution integral technique; the entire domain is shown. B: the object's trajectory, and the definition of drift angle α. The red dots here and in panel C mark the selected beginning of the dynamically steady state, the red dashed line is the linear fitting of the trajectory. C: time series of the object tilt angle θ, the red dashed line is the time average of θ over the period of the dynamically steady state. D: a zoom in view showing the vorticity field around the object. The object is marked by the contour $\epsilon = 0.5$.

3.9 Extension to Temperature Equation

We briefly show that the solid forcing term $\epsilon(\boldsymbol{u}_s - \boldsymbol{u})/\Delta t$ can be applied to temperature equation as well. By analogy the "solid thermal forcing" term is $\epsilon(T_s - T)/\Delta t$. Of course

this method does not handle the flux boundary which requires the specification of gradient instead of the value. But if the problem only has Dirichlet boundary conditions, the method presented in this chapter applies directly. We test this idea against the heat transfer from a 2D cylinder to the passing flow. The correlation between the average Nusselt number and the Reynolds number by Zhukauskas and Jakob is

$$\overline{Nu} = CRe^m Pr^{1/3}$$

where $Re = UD/\nu$, U is flow speed, D is the cylinder diameter. The comparison of our calculation with the correlation is given in the figure below. The relative is a few percent.

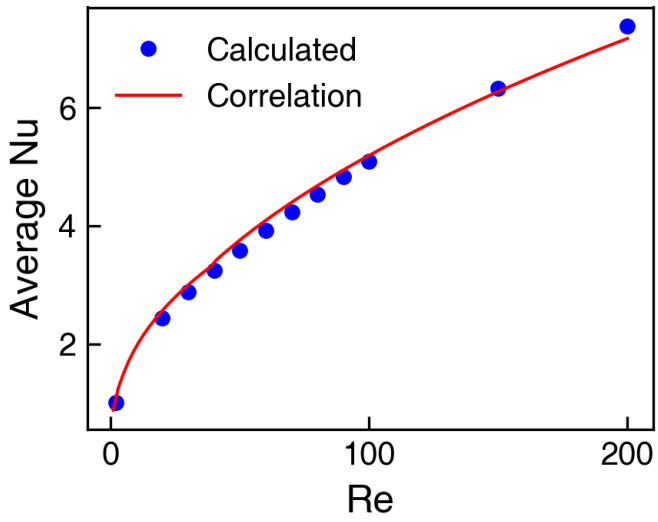

Figure 3.13: Comparison of calculated average Nusselt number with Zhukauskas-Jakob correlation, for Reynolds numbers between 2 and 200

3.10 Conclusion

An accurate and robust algorithm for solid volume fraction calculation is developed under the context of immersed boundary method in computational fluid dynamics (CFD). The signed distance field, which has been widely used in computer graphics for geometry representation, is applied to calculate the fractional volume of any CFD cell cut by the solid boundary and thus yield the solid volume fraction. The application of signed distance field

for this purpose avoids the complexity of geometry intersection test which is commonly practiced when the solid boundary is implicitly defined. The present algorithm proves an accuracy of second order in space. Numerical simulations show that the accurate calculation of solid volume fraction significantly improves the overall accuracy of the immersed boundary method.

Signed distance field and the volume fraction based immersed boundary method together enables a very uniform interface of different geometries. The only information a geometry needs to provide is a function that returns its signed distance field. Since no intersection tests are needed, the complexity of the geometry itself incurs little extra cost, in terms of both algorithm modification and computational cost. Or from a software engineering point of view, the signed distance "add a layer of indirectness" to the algorithm in the sense that it hides behind a uniform representation of highly different geometries. Almost arbitrarily complex geometries can be created with ease through Boolean operation or other spatial transformations of elementary geometries, enabled by the properties of signed distance.

The convenience brought by signed distance is partially explained by the fact that it encodes more information compared with a straightforward surface definition of the form $f(\boldsymbol{p}) = 0$. Taking the unit circle as an example, while $x^2 + y^2 - 1 = 0$ defines the circle, it only provides a true or false answer to the basic "point on circle" test. On the other hand, the signed distance representation $\sqrt{x^2 + y^2} - 1$ not only does this, it tells if the point is inside or outside, quantitatively.

Chapter 4
Coupled Immersed Boundary and Discrete Particle Simulation

4.1 Introduction

In this chapter we study the granular system where hydrodynamics plays an important role. Those so called gas-solid or liquid-solid systems is ubiquitous in nature (sand storm, sedimentation, etc.) and industry (pneumatic transport, fluidized bed, slurry). Computational models with different levels of resolution of the solid component have been developed.

1. At the coarse level, the solid is considered as a continuous phase, just like the fluid. Mixture or two fluid model belong to this category. In those models the solid exist on the same mesh as the fluid.

2. In the intermediate level, each solid, often refereed as a particle, has its own identity. In other words, a Lagrangian approach is used. This approach is further classified according to the number of interactions taken into account. If the fluid affects the particle, but not vice versa, we have an one-way coupling. If the particle affects the fluid as well, we have two-way coupling. If further the particles affect each other (collision or long ranged force), we have four-way coupling. Lagrangian tracking of passive particles, for example, only considers the one-way coupling. When four-way coupling is used, the model is often referred as discrete particle method or DPM. Methods at this level of resolution still relies on certain empirical input, the main one being the drag of a particle by fluid when calculating the effect of fluid on solid. To calculate the effect of solid on fluid for two and four way couplings, the solid volume occupation and the drag on the fluid need to be averaged and "mapped back" onto the Eulerian mesh, where the fluid phase exists. A proper average requires the mesh cell to be greater than the particle size.

3. At the finest level, the flow field around every solid is resolved explicitly, and no empirical input is necessary. Methods of this category are all considered as direct numerical simulation (DNS), even thought the detailed implementation can vary greatly. The immersed boundary method presented in the previous chapter is one such method, other DNS methods could use different approaches like dynamic mesh, overset mesh, fictitious domain, etc. In terms of mesh resolution, DNS methods require mesh cells smaller than the solid (see last chapter for examples). For this reason DNS is computational very expensive and currently limited mostly to a few thousand particles and laminar flow regimes.

In this chapter we are concerned with problems where the solids consist of both large object and small particles, and they coexist in a fluid environment. An example of such system is the well known experiment of dropping a steel ball into an open container of powders (some example videos are available at `http://jfi.uchicago.edu/~jaeger/group/granular2/jets.html`). Clearly a DNS approach is impractical considering the amount of particles, on the other hand, the DPM approach can not handle the large object at that would require a grid cell size several times the object size. Basically due to the scale difference of the solids, we have the following restriction on the grid resolution: the grid cell need to be smaller than the large object, while greater than the small particle. For this reason we need to use both DNS and discrete particle treatment.

There are few published works on the coupled immersed boundary discrete particle simulations. Xu et al. [2013] treated the large object (a sphere) by sampling it with "markers", following Uhlmann [2005]. Each marker exerts a force on the fluid such that the nonslip condition on the object is met. The trajectory of the sphere was calculated and compared with experimental data. Sun and Sakai [2015] combined different algorithms to simulate a system consist of small particles (by DPM), gas *and* liquid (by VOF), and wall boundaries (by IBM). Their focus was on the interaction between particles and the interface between two liquid phases. The walls treated by immersed boundary method remains static.

4.2 Solid Volume Fraction Calculation

4.2.1 Heuristic

We use the same direct forcing, volume fraction based immersed boundary formulation as in the previous chapter. As was pointed out earlier, the solid volume fraction ϵ is a multiplicative factor of the forcing, so any error of it directly transfer to the fluid solid interaction force. We proposed the signed distance function based volume fraction calculation. Here however, we explore another method that based on a simple heuristic that is cheaper to implement and calculate, and still with good accuracy.

Consider how the simplest step method handles a cell's solid volume fraction: it assigns $\epsilon = 0$ if the cell center is outside of the geometry, and assigns $\epsilon = 1$ if it is inside of the geometry. A direct extension is then to use more sampling points. But instead of generating new points in space, we use points from the mesh construction itself, i.e., nodes, face centers, and cell centers. Apparently the node should have different weight from the face center, and from the cell center. This idea was proposed before by Agrawal et al. [2009], although it was unclear how the weights were assigned in their method.

To assign weights to those different type of points in 2D, we consider an infinite plane that is tiled by a single template (Figure 4.1). The template centroid corresponds to the sampling point (node, face center, and cell center). The template's size, shape, and orientation depend on the distribution of sampling points and need not to be aligned with the CFD grid. This method generalizes to any unstructured mesh by introducing the concept of Voronoi spatial partition, but is unlikely to have practical value as different cells have different shapes and each would have to be treated specifically. In our case however, where all mesh cells are square, the tile is easily obtained. The weight of a point is then the fractional area of its tile falling inside the cell, divided by the area of the cell.

Figure 4.1 shows two such heuristics. They may be referred as "five-point method" and "nine-point" method, depending on their number of sampling points. Their volume

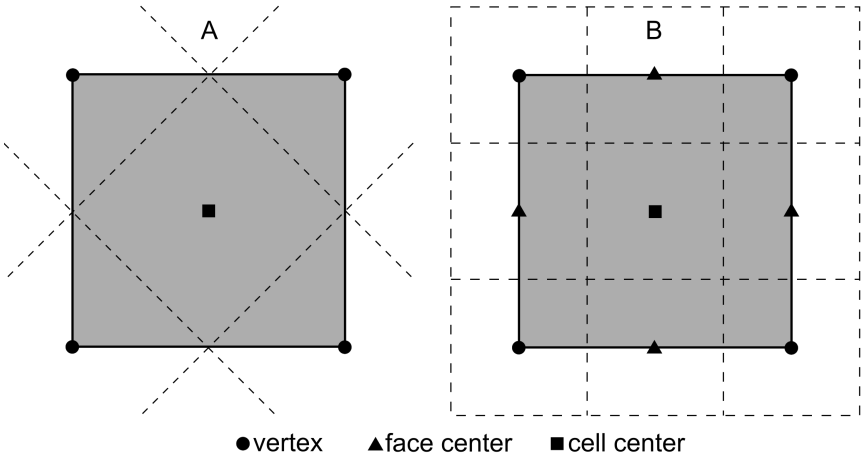

Figure 4.1: Volume fraction calculation heuristics.

fraction formulas can be derived easily. For the five point method we have:

$$\epsilon = \frac{1}{8}\left(v_1 + v_2 + v_3 + v_4 + 4c\right)$$

and for the nine point method:

$$\epsilon = \frac{1}{16}\left(v_1 + v_2 + v_3 + v_4 + 2(e_1 + e_2 + e_3 + e_4) + 4c\right)$$

where $v_i/e_i/c$ represent the contribution from nodes, edge center, and centroid, respectively. It equals 1 or 0 depending if that point falls inside the solid. The generalization to 3D is straightforward.

4.2.2 Validation

We use a 2D case to validate the solid volume fraction heuristics. An $[0, 1] \times [0, 2]$ domain is used, a circle of radius 0.2 is allowed to fall freely from $[0.5, 1.5]$. Its density (2) is double that of the fluid (1). The fluid kinetic viscosity equals 0.01. The gravity constant equals 9.8 and points downward. The time series of the vertical position and velocity is shown in Figure 4.2. The five and nine point method improves significant compared with the 0-1 method.

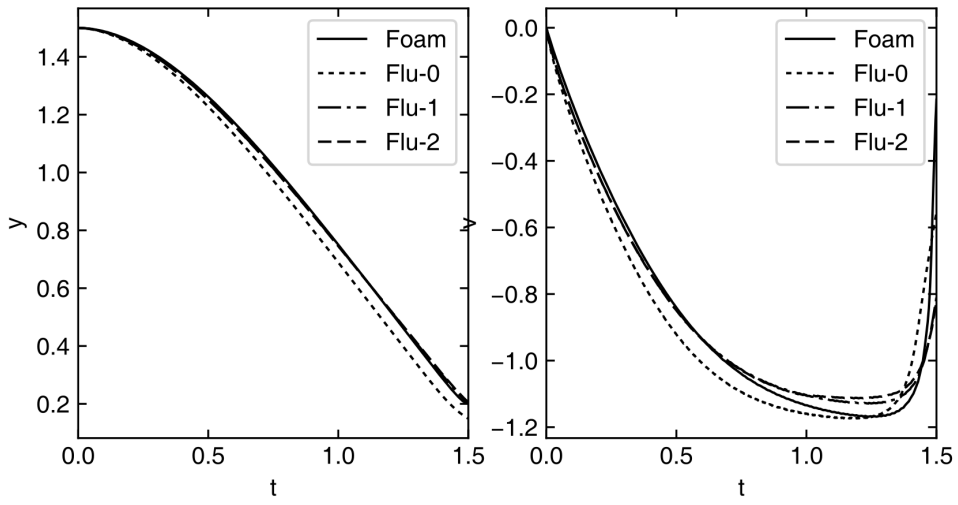

Figure 4.2: Validation of the solid volume fraction calculation, using the 0-1 method (Flu-0), five point method (Flu-1), nine point method (Flu-2). Those three calculations by Fluent is compared with the SDF based OpenFOAM calculation (Foam).

4.3 Void Rising Experiment

We study the rising of a single void in a two dimensional granular bed to understand the effect of particle size. Three particle diameters, 212 micron, 106 micron, and 53 microns are used. This is the only parameter we vary. Their corresponding cases will be referred as large (particle size), medium, and small, respectively. All other parameters including the initial region of packing and material properties are the same. The initial condition is shown in Figure 4.3. The void is circular initially, with diameter equals $10mm$, and center locates at $(17.5mm, 17.5mm)$. The color in Figure 4.3 represents the solid volume fraction clipped to $[0, 0.54]$. The same color range apples to all following figures in this section.

Air bubbles rising in the fluidized bed is a familiar phenomenon. We didn't find similar setups in the literature. The mostly related work in terms of setup is that by Loranca-Ramos et al. [2015], where an air balloon is buried beneath sand and punctured. Since their main interest is crater formation and shape, the balloon is buried shallowly and the air bubble left after the puncture quickly erupts at the free surface without noticeable rising.

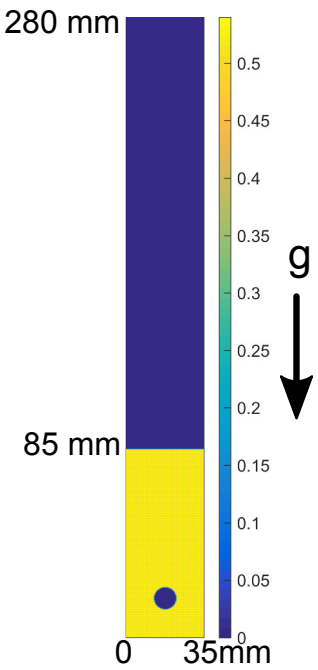

Figure 4.3: Initial condition for the bubble rising experiments. A circular void is created at $(17.5mm, 17.5mm)$, with diameter equals $10mm$.

The following three figures, each consisting four snapshots of the system, clearly show the effects of particle sizes. For the large particles, the void can not hold its shape and collapses quickly after initialization. In fact, this is similar to if there is no air at all, i.e., pure granular dynamics. The reason is first, larger particles are less susceptible to the air flow due to greater mass; second, while the initial packing fraction by our square lattice arrangement is independent of particle size. The permeability of the disrupted granular bed does depend on it—and larger particles gives higher permeability according to the Carman-Kozeny relation [P. C. Carman, 1957]. This facilitate the venting of the air, it can be seen that basically no air is left by time $0.15s$.

When the particle diameter is reduced by half (medium particle), the dynamics is dramatically different. The void behaves very similar to a bubble. It maintains its integrity and oscillate when rising, as if there is an effective surface tension. In addition, the flow pattern—as shown by the streamlines—forms the well know convective cell pattern. Due to the two dimensionality, the bubble is eventually split into halves (t = 0.2 panel). This

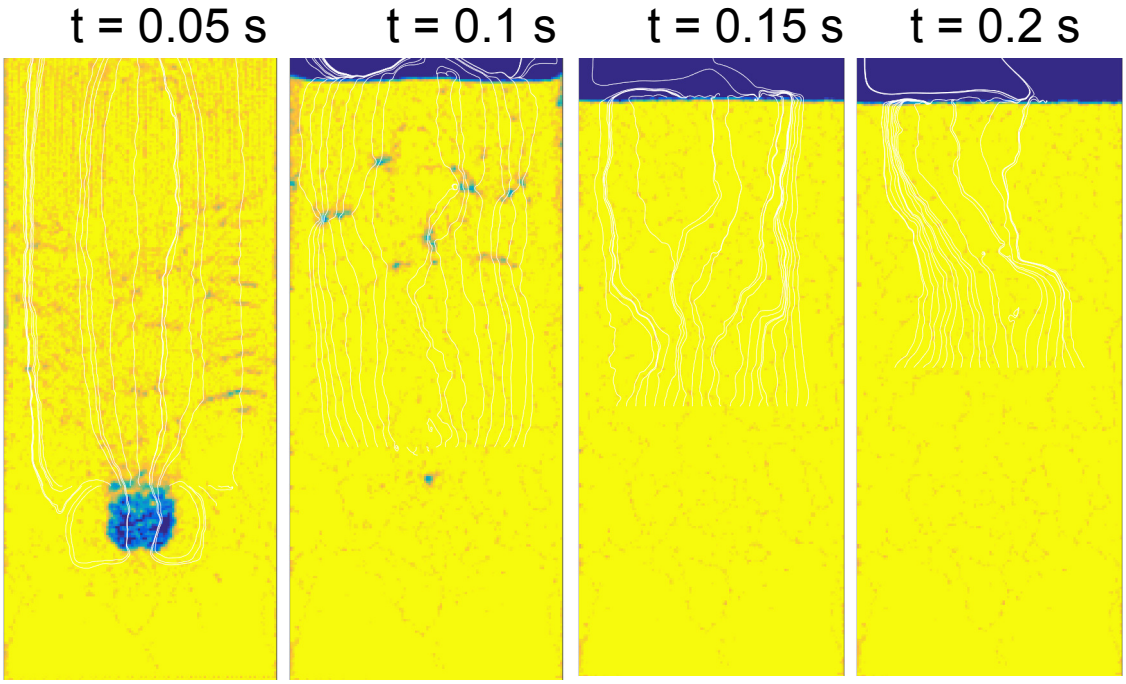

Figure 4.4: Four snapshots of rising void. The spatial range of the panels are $[0, 35mm] \times [0, 80mm]$. The granular material is mono-dispersed with particle diameter equals 212 micron.

is one uniqueness caused by dimensionality. The breakup behavior could be different in three dimensions. Compared with large particle case, a medium particle only has 1/8 the mass. In addition, the gaps between the particles are smaller so the gas escapes harder.

For the small particle case. We observe similar bubble like behavior initially. For example, the $t = 0.05$ snapshot in Figure 3.6 is very similar to the corresponding panel in Figure 3.5. However, the evolution afterwards is different in that the bubble quickly breaks up. On the other hand, the rising speed of the bubble is not much different from the medium case. As a result, by analogy with the gas-fluid system, we may say the bubble has a smaller effective surface tension.

In all cases, there are a population of small bubbles rising near the top of the bed. The reason is we started from a square lattice of particles. This is a stable configuration if there was no bubble. Due to the perturbation of the bubble, this cubic lattices collapse and compact to a denser packing. For example, the rightmost panel in Figure 3.4 is near steady state: the voidage at the top is clearly larger than the initial bubble size. Plot of

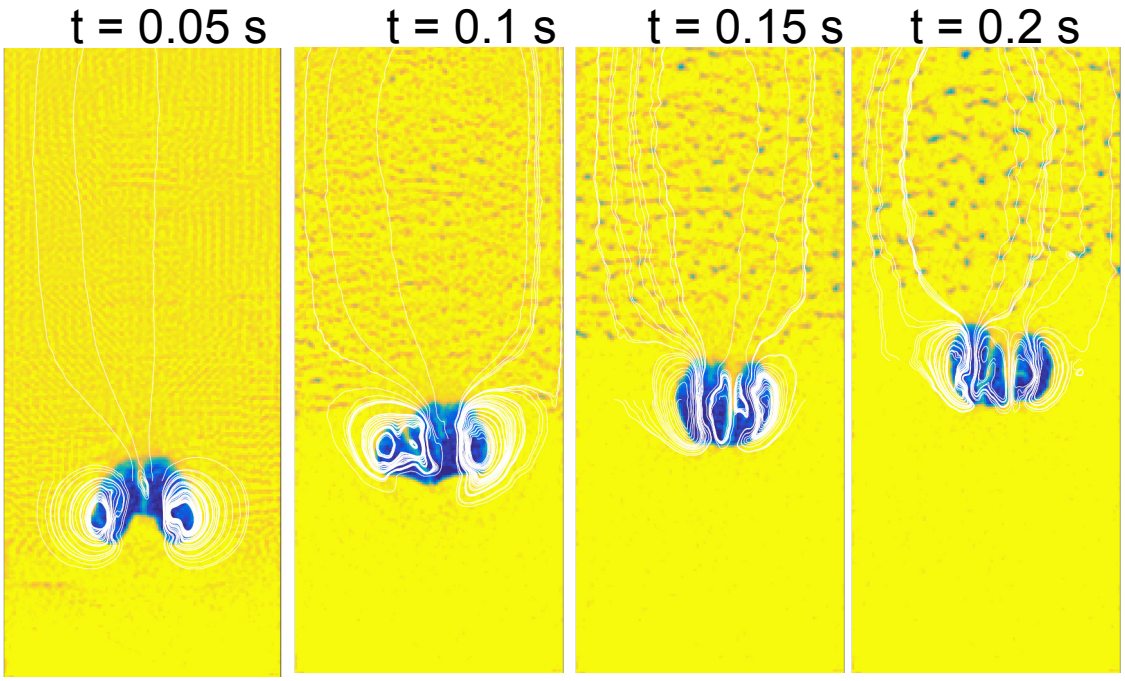

Figure 4.5: Four snapshots of rising void. The spatial range of the panels are $[0, 35mm] \times [0, 80mm]$. The granular material is mono-dispersed with particle diameter equals 106 micron.

the particles clearly shows that the end configuration involves patches of hexagonal lattice, similar to the one discussed in Chapter 2. For the next IBM-DPM simulation, we use the medium sized particle (105 micron), as it fluidizes relatively easily, and computationally cheaper than the small particle (which will quadruple the number of particles and the computational resource).

4.4 Outline of Implementation

4.4.1 Discrete Particle Modeling

The DPM algorithm was presented in detail in Wu et al. [2014] and previous dissertation by Dr. Oladapo Ayeni[1]. It is summarized below (based on [Wu et al., 2014]).

[1]LSU Doctoral Dissertations: Application of Discrete Element Method and Computational Fluid Dynamics to Selected Dispersed Phase Flow Problems

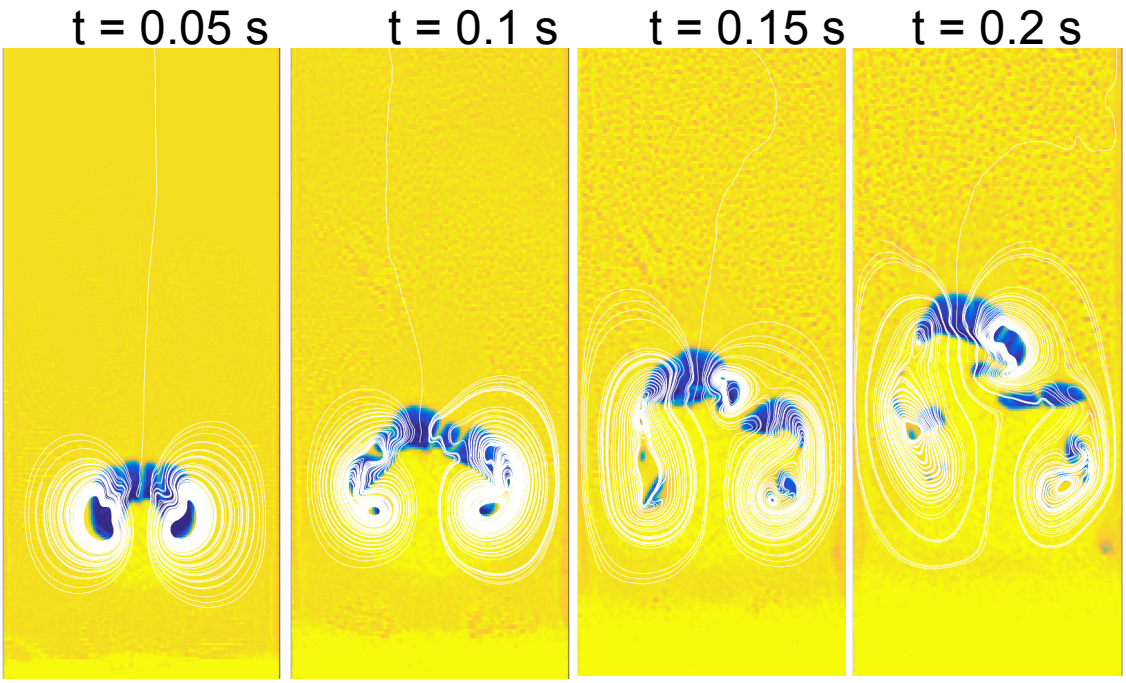

Figure 4.6: Four snapshots of rising void. The spatial range of the panels are $[0, 35mm] \times [0, 80mm]$. The granular material is mono-dispersed with particle diameter equals 53 micron.

The phase averaged (fluid phase and particle phase) equation is

$$\begin{aligned} &\frac{\partial}{\partial t}(\epsilon\rho_f) + \nabla\cdot(\epsilon\rho_f \boldsymbol{u}_f) = 0 \\ &\frac{\partial}{\partial t}(\epsilon\rho_f \boldsymbol{u}_f) + \nabla\cdot(\epsilon\rho_f \boldsymbol{u}_f\boldsymbol{u}_f) = -\epsilon\nabla p + \nabla\cdot(\epsilon\boldsymbol{\tau}_f) + \boldsymbol{S}_f + \epsilon\rho_f\boldsymbol{g} \end{aligned} \tag{4.1}$$

Where ρ_f is the fluid density, ϵ is the void fraction of the fluid phase, and $\boldsymbol{\tau}_f = \mu(\nabla\boldsymbol{u}_f + \nabla\boldsymbol{u}_f^T) + \frac{2}{3}\mu(\nabla\cdot\boldsymbol{u})\boldsymbol{I}$ is the stress tensor. The $\boldsymbol{S}_f$ term represents the fluid-particle interaction force. If only drag force is considered, then $\boldsymbol{S}_f$ is a summation of the drag force contributed by all particles

$$\boldsymbol{S}_f = \frac{1}{V}\int_V \sum_{k=1}^{N_p} \boldsymbol{F}_{d,k}\delta(\boldsymbol{x} - \boldsymbol{x}_{p,k})dV$$

where N_p is the total number of particles in the integral volume V, δ is the Dirac delta function. The drag force on each particle is $\boldsymbol{F}_d = V_p\beta(\boldsymbol{u}_f - \boldsymbol{u}_p)/(1-\epsilon)$, where V_p is the particle volume, $\boldsymbol{u}_p$ is the particle velocity. β is an empirically defined momentum exchange

correlation, we use the Gidaspow's model [Gidaspow, 1994]:

$$\beta = \begin{cases} 150\dfrac{(1-\epsilon)^2\mu}{\epsilon d_p^2} + 1.75(1-\epsilon)\dfrac{\rho_f}{d_p}|\boldsymbol{u}_f - \boldsymbol{u}_p|, & \epsilon \leq 0.8 \\ \dfrac{3}{4}C_D\dfrac{\epsilon(1-\epsilon)}{d_p}\rho_f|\boldsymbol{u}_f - \boldsymbol{u}_p|\epsilon^{-2.65}, & \epsilon >^{\circ} 0.8 \end{cases}$$

d_p is the particle diameter. C_D is the drag coefficient for a single particle define as

$$C_D = \begin{cases} \dfrac{24}{Re_p}(1 + 0.15Re_p^{0.687}), & Re_p \leq 1000 \\ 0.44 \quad Re_p > 1000 \end{cases}$$

Re_p is the particle Reynolds number based on local relative velocity

$$Re_p = \frac{\epsilon\rho_f d_p|\boldsymbol{u}_f - \boldsymbol{u}_p|}{\mu}.$$

The key idea to solve the governing equation 4.1 is to transform it into a single phase form through algebraic manipulation.

$$\begin{aligned} &\frac{\partial}{\partial t}(\rho_f) + \nabla\cdot(\rho_f\boldsymbol{u}_f) = S_c \\ &\frac{\partial}{\partial t}(\rho_f\boldsymbol{u}_f) + \nabla\cdot(\rho_f\boldsymbol{u}_f\boldsymbol{u}_f) = -\nabla p + \nabla\cdot(\boldsymbol{\tau}_f) + \boldsymbol{S}_m + \rho_f\boldsymbol{g} \end{aligned} \tag{4.2}$$

where $S_c = \frac{\rho_f}{\epsilon}(\frac{\partial\epsilon}{\partial t} + \boldsymbol{u_f}\cdot\nabla\epsilon)$, $\boldsymbol{S}_m = S_c\boldsymbol{u}_f + (\boldsymbol{S}_f + \boldsymbol{\tau}\cdot\nabla\epsilon)/\epsilon$. This transformation turns the effect of solid phase into source terms, and makes it very convenient to use. The source term can be defined and applied to various existing codes, both commercial ones like FLUENT through user defined functions or "UDF", and open source solver like OpenFOAM by writing extra codes. More implementation details like the finite volume discretization, the momentum exchange between the particles and fluid, should be found in [Wu et al., 2014].

Since our immersed boundary method is implemented by adding source terms to the momentum equation, it can be merged naturally with the in-house DPM code [Wu et al., 2014]. The most important technical details regarding the immersed boundary extension

are summarized below.

1. Large sphere immersed boundary implementation, using the forcing formulation introduced in the last chapter.

2. Large sphere-small spherical particle contact. The main modification is each small particle needs to store the tangential deformation history of its contact with the large sphere.

3. MPI parallelization. The simplest parallelization strategy is used, in that each processor keeps a copy of the large sphere. This avoids complex communication and synchronization. This is quite affordable when there is only a few large objects (in our case only one). To synchronize the force acted on the large sphere by small particles and by the fluid, a MPI barrier is inserted before the time evolution, then a parallel sum reduction followed by a broadcast is sufficient to update all processors' copy of the large object to the correct total forcing.

4. All extensions are done in Fluent UDF using a mixture of ANSI-C and object oriented C++.

The main limitation of the current approach is that the volume exclusion effect of the large object is not considered. This means when a mesh cell contains both the large object and small particles, the fluid in that cell is ideally excluded by both solids. However, we only considered the solid volume fraction contributed by small particles. The large object's volume fraction contribute only through the direct forcing formulation. This is not likely to affect our granular impact case (next section) because when all three (small particle, large object, and fluid) are present in a mesh cell, the dominant dynamics there is collisions between the solids, especially considering the large density difference between the solids and the air. In less energetic scenarios however, or when the fluid has comparable density as the solids, an accurate way of handling volume fraction will be important.

4.5 Sphere Impact Experiment

4.5.1 Introduction

Impact of an object onto a bed of granular material is of wide interest, including crater formation Ambroso et al. [2005], drag laws of object moving through granular material Hou et al. [2005], Pacheco-Vazquez et al. [2011] etc. Here we only briefly review the literature where the object penetrates deeply into the granular material, and the formation of granular jet is the primary interest.

The experiment was first done by Thoroddsen and Shen [2001] using a lead sphere and different sized, mono-disperse spherical glass beads. A scaling of the maximum granular jet height h_j was proposed as $h_j/d_s \propto d_s^2/d_p^2 \cdot V/\sqrt{gd_s}$, where d_s is the sphere diameter, d_p is particle diameter, V is the impact velocity, and g is the gravity constant. An implication from this scaling is the particle diameter has significant effect on h_j. Mikkelsen et al. [2002] experimented by dropping a steel sphere (diameter is 25 mm) into loose bed of fine sands (average diameter about 40 micron). The loose packing is achieved by blowing air from bottom of the bed and slowly turning it off to avoid compaction. A two dimensional setup from the same paper [Mikkelsen et al., 2002] reveals that as the sphere penetrating the bed, the collapse of the sand behind it traps an air bubble. The collapse also creates an upward jet directly into the air above the granular bed, and a downward jet into the air bubble. The bubble rises upwards and eventually erupts at the surface of the sand bed. A follow-up study Lohse et al. [2004] measured the jet height. Contrary to the observation in [Thoroddsen and Shen, 2001], the jet height exceeds the initial height of the sphere, possibly due to the looser packing and the smaller particle size. Result from a discrete particle simulation was also presented. No details however were given about its formulation, especially how the interaction between the sphere and air is handled. Their simulation (FIG. 3 therein) does not correctly capture the trapped air bubble shown in their previous experiment Mikkelsen et al. [2002].

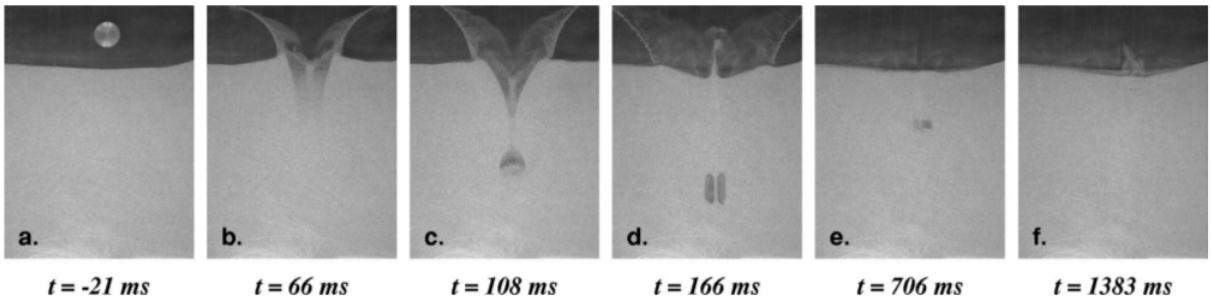

Figure 4.7: The figure 2 extracted from Mikkelsen et al. [2002]. Shown are snapshots of a quasi-2D setup of a loose sand bed impacted by a steel cylinder.

Royer et al. [2005] applied X-ray radiography and obtained an (2D projected) internal view of the granular jet development. They find the granular jet has two parts: a thin air-independent part at the top, below which is a thicker air-dependent jet. The thin jet is caused mainly by the collision of grains, while the thick jet is caused by the trap and eruption of the air bubble. They explored the effect of ambient air pressure P on the jet velocity v. Based on an energy argument that the pressure of the air bubble and a small compression δV provided the jetting energy, a relation $v \propto \sqrt{P - P_0}$ is proposed, where P_0 is a reference pressure. This relation fits the experimental data nicely.

4.5.2 Result

A sphere initially locates $80mm$ above the bed falls vertically downwards at a speed of $2.3m/s$. Figure 4.8 shows when the cylinder collides with the particles. The impact of the cylinder compacts the particles and increasing their volume fraction (represented by the lighter yellow color). A front between the compacted and the unperturbed regions is observed propagating ahead of the cylinder. After the front reaches the side walls (Figure 4.8, last panel), the particles climb up the wall (Figure 4.9). This forms a slope which eventually collapses under gravity, as shown in the first panel of Figure 4.10, where the reverse of the granular flow on the slope has disrupted the air velocity field. The left and right slopes collapse and merge almost right above the cylinder. Air is displaced upwards as shown by the vector plot. The slopes continue collapsing towards the center of the domain in Figure 4.11, a jet is formed due to this collision. At the same time, an air bubble is

formed from the air entrained by the cylinder and starts rising. Figure 4.12 and Figure 4.13 show the subsequent evolution of the jet and the bubble.

In this simulation, we only observed the upward jet (Figure 4.7, panel d). Below is some possible causes.

1. The particles has relatively large friction (friction coefficient equals 0.3). Considering that we are dealing with a two dimensional setup, the friction, as a surface force, has much stronger effects (in terms of surface/body force ratio) compared with a three dimensional setup. It quickly damps the cylinder motion and reduces its penetration depth.

2. The wall effect. The side walls start to affect the particle motion quickly after the cylinder impact. The wall confinement causes the particles displaced by the cylinder to move upwards, which give rise to higher slopes. The slope shape directly affects its later collapse. In both 4.7 and the x-ray radiography by Royer et al. [2008], a larger air bubble is trapped as the slope collapses. This bubble allows the space for the formation of a downward jet. It later rises and erupts at the free surface of the granular bed. In our case however, the merge of the slopes only traps a very small amount of air, which is insufficient to generate a secondary jet.

In light of the above speculation, we explored another setup (Case II). Several modifications were made. First the particle friction coefficient is reduced from 0.3 to 0.05. Second, the cylinder diameter is reduced from $1cm$ to $6mm$. Third, the cylinder starts $20mm$ above the granular bed with an initial downward velocity of $3m/s$ to reduce the computational time. Figure 4.14 to 4.17 shows snapshots of this setup. The initial stage after the impact shown in Figure 4.14 is similar to the previous case (Figure 4.9) in terms of the front of compaction and formation of slopes. The collapse after that is quite different (Figure 4.15): the two slopes merge at a distance above the cylinder and enclose a much more noticeable air bubble behind it. The collision of the two slopes creates the pair of

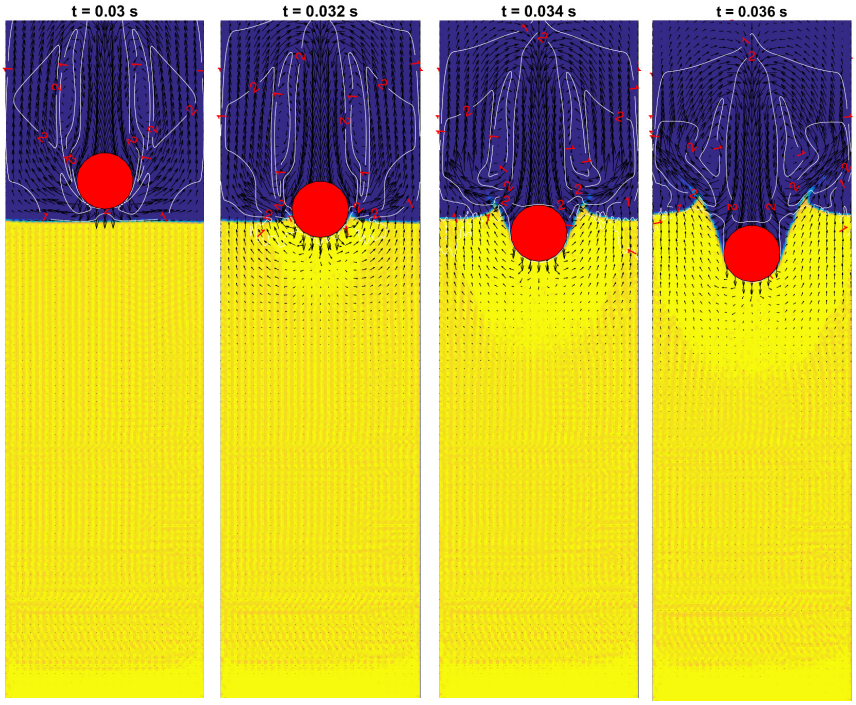

Figure 4.8: Snapshots from $t = 0.03s$ to $t = 0.036s$.

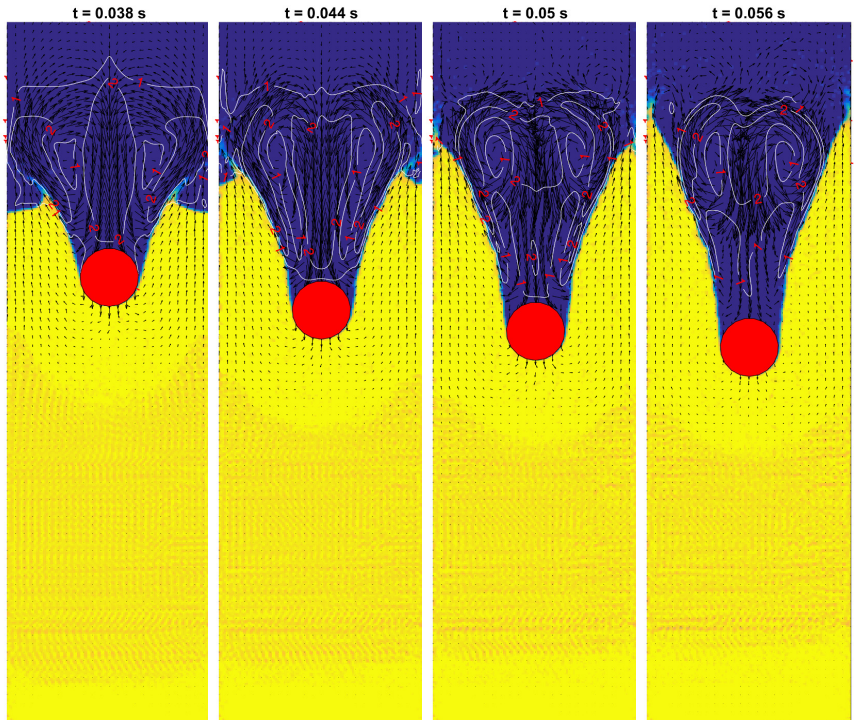

Figure 4.9: Case I. Snapshots from $t = 0.038s$ to $t = 0.056s$.

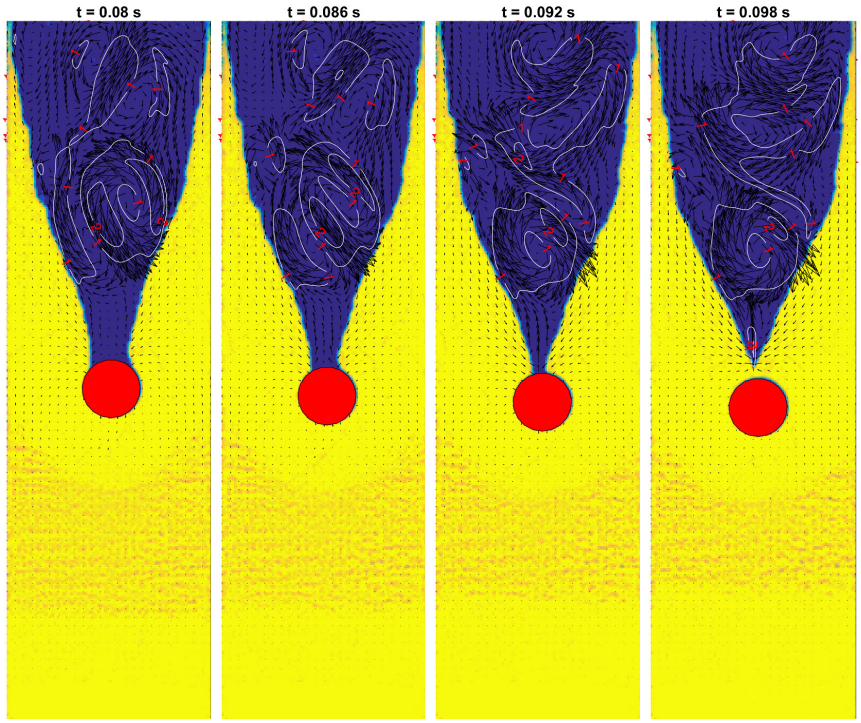

Figure 4.10: Case I. Snapshots from $t = 0.08s$ to $t = 0.098s$.

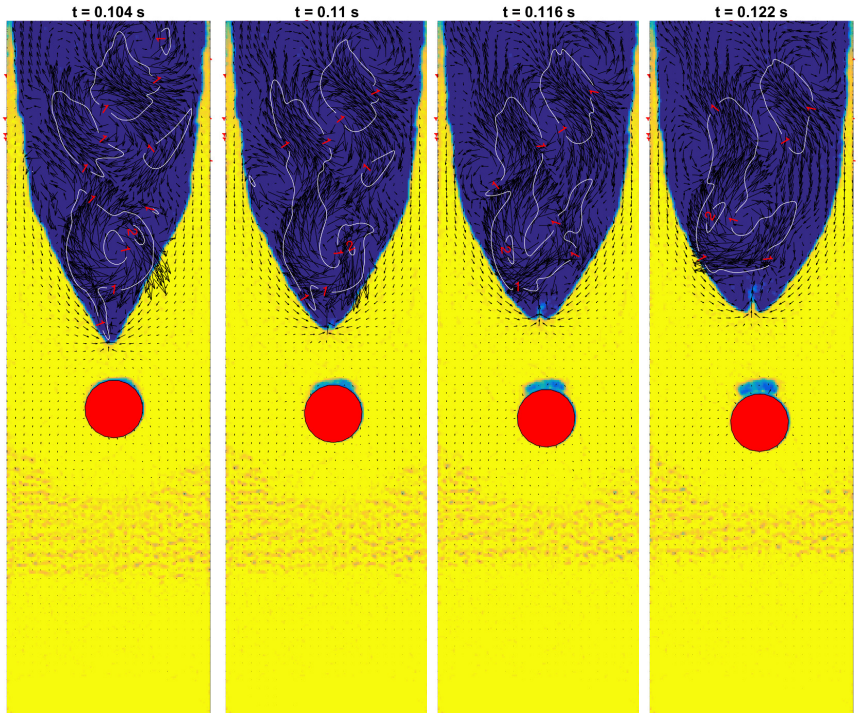

Figure 4.11: Case I. Snapshots from $t = 0.104s$ to $t = 0.122s$.

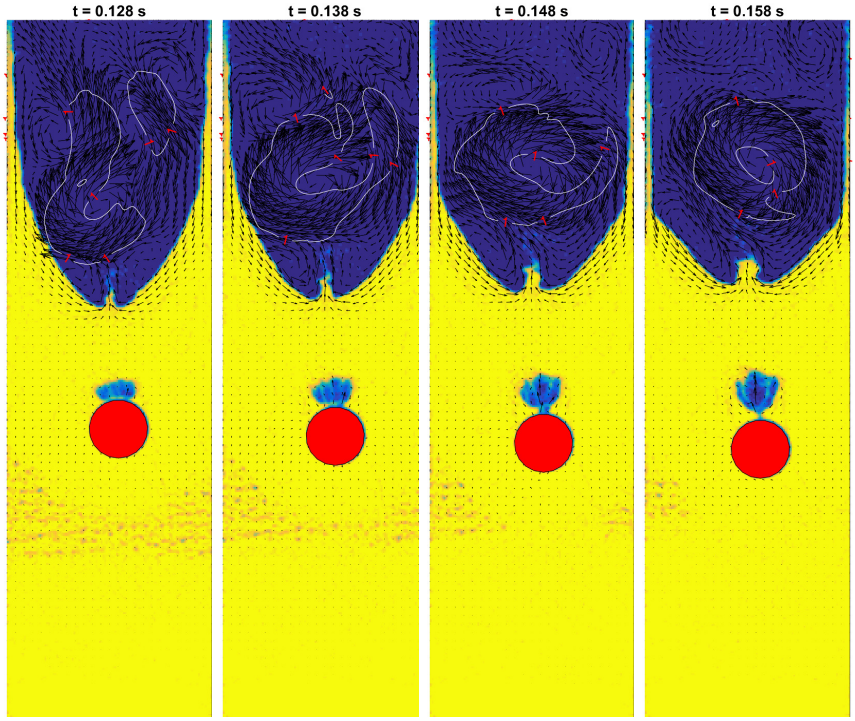

Figure 4.12: Case I. Snapshots from $t = 0.128s$ to $t = 0.158s$.

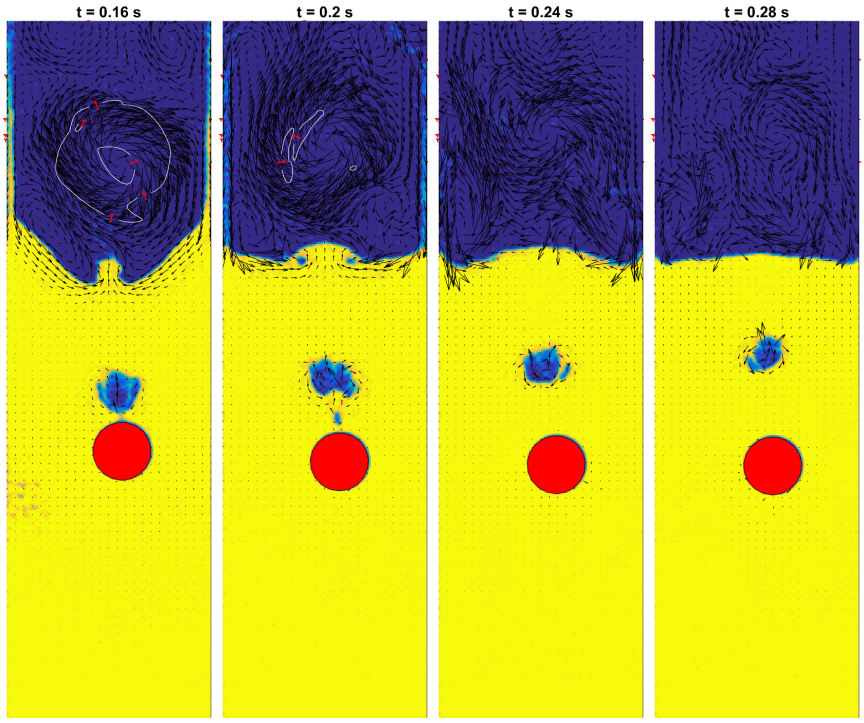

Figure 4.13: Case I. Snapshots from $t = 0.16s$ to $t = 0.028s$.

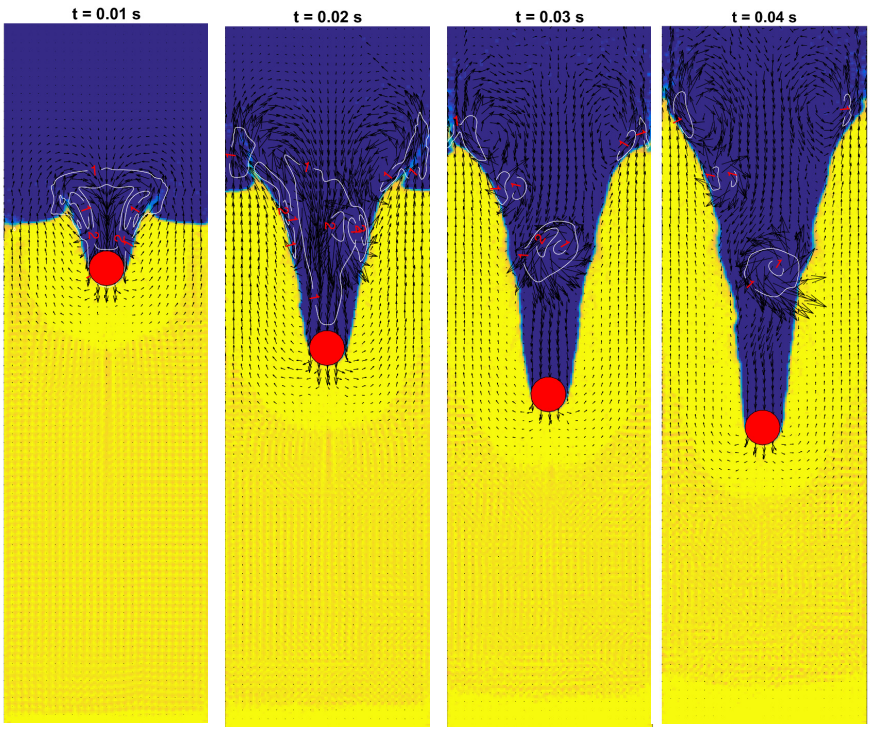

Figure 4.14: Case II. Snapshots from $t = 0.01s$ to $t = 0.04s$.

upward and downward jets (Figure 4.16). The downward jet later splits the bubble into two, which rise and dissolve into the granular bed (Figure 4.16 and 4.17). This sequence of events is similar to 4.7, and qualitatively validated our implementation.

The computational efficiency is OK, for all cases in this chapter, a mesh with 120×960 cells is used, and the number of particles is about 250,000. Using a fluid time step of 1×10^{-4} and a particle time step of 1×10^{-6}, it took about 36 hours to simulate 0.3 seconds on LSU SMIC using 40 cores (2 nodes, each has 20 cores).

4.6 Conclusion

This chapter presents an implementation of a coupled immersed boundary-discrete particle method. The main simplification is the neglect of the volume fraction contributed by the solid in a CFD cell, when the fluid and small particles also present. A simple parallelization strategy, which that can be easily implemented, is applied.

We were able to produce qualitatively correct result comparing with the experiment after reducing the wall effect. A interesting observation that has not been noted before is

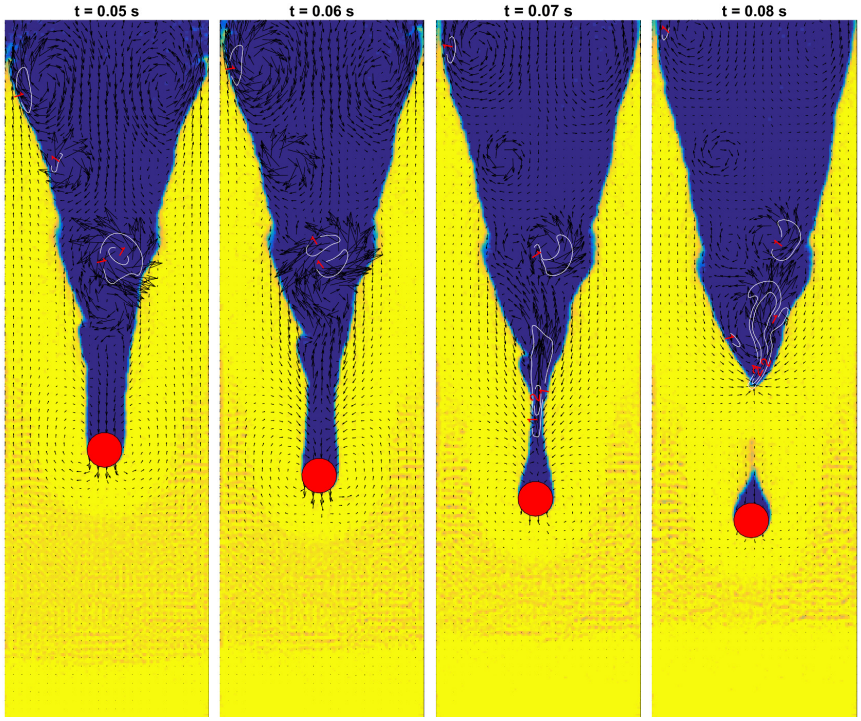

Figure 4.15: Case II. Snapshots from $t = 0.05s$ to $t = 0.08s$.

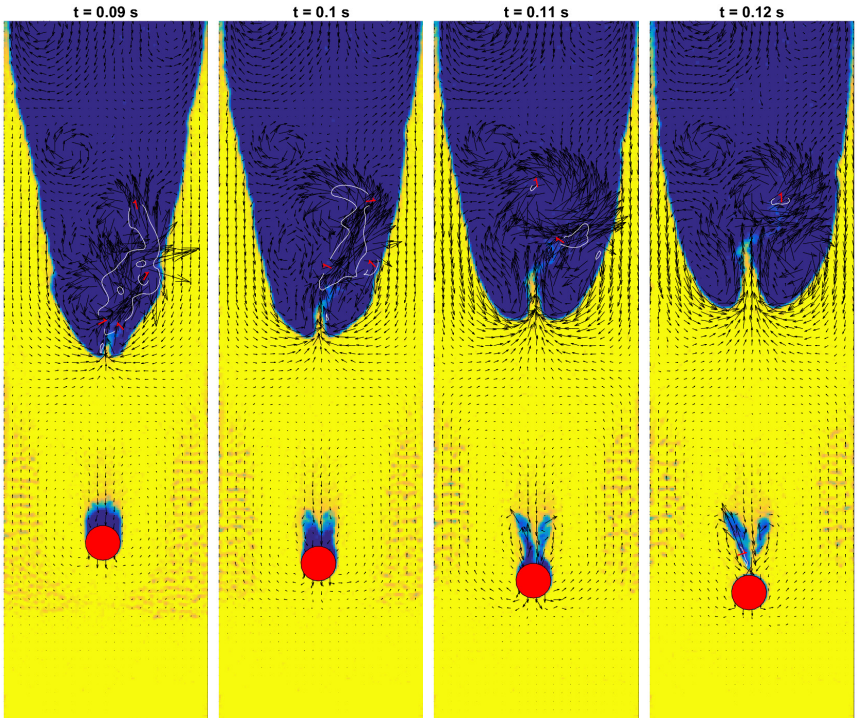

Figure 4.16: Case II. Snapshots from $t = 0.09s$ to $t = 0.12s$.

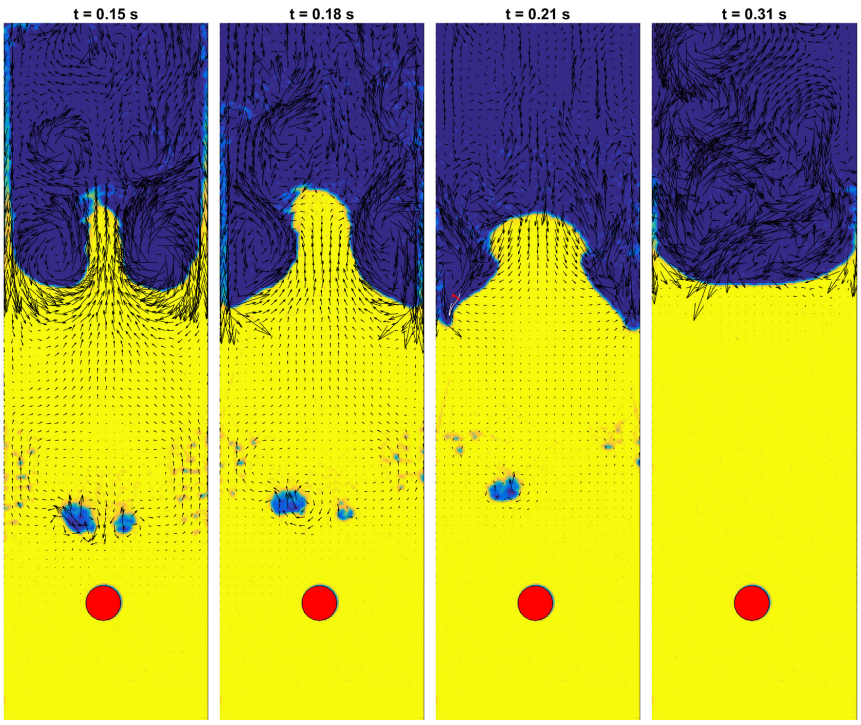

Figure 4.17: Case II. Snapshots from $t = 0.15s$ to $t = 0.31s$.

the energetic jetting could be completely eliminated in a two dimensional setup with very narrow walls. The "hydrodynamic pressure" Lohse et al. [2004] of the displaced grains is quite different from one where very wide container is used (no wall effect), such that the slopes merges immediately above the cylinder, trapping only a very small amount of air.

Chapter 5
Two Dimensional Thermal Wave Driven Flow

5.1 Introduction

Micro fluidic devices are beginning to fulfill their promise in a number of applications areas such as in micro Total Analysis[Reyes et al., 2002], medical diagnostic devices, physical, chemical and biological analytical processes [Stone et al., 2004] and in mono sized droplet generation [Shi et al., 2014]. With such developments, an interest in understanding fluid transport or pumping by non-mechanical means is also drawing increasing attention. Several alternate mechanisms, such as capillary, acoustic, electrokinetic and peristaltic mechanisms [Shi et al., 2014, Eldabe et al., 2007, Muthuraj and Srinivas, 2010] could provide new non-mechanical means of pumping fluids through micro devices in small amounts. Laser and Santiago [2004] review the development over a 25-year period of micro scale devices for pumping small amounts of fluids in microfluidic devices. Stone et al. [2004] provide a much broader overview of microfluidic devices toward a Lab-on-a-chip.

In this chapter we propose an alternate mechanism for non-mechanical pumping of fluid that is well suited when thermal effects are naturally present in microfluidic device. The mechanism of transporting fluid by density gradients induced by boundary heating has been speculated in the past in a completely different context. In the 16th century Dr. Halley hypothesized that the global atmospheric circulation is caused by the diurnal heating from the sun as the earth rotates [Halley, 1686]. Although incorrect, this idea inspired the "moving flame experiment" by Fultz [1956], and later by Stern [1959]. The observation that vertical buoyancy force could generate net horizontal momentum attracted many interest. Stern [1959] gave the first theoretical analysis of this problem in the zero Prandtl limit (infinite thermal diffusivity) and with long wave assumptions. It was shown that horizontal momentum could be created by the Reynolds stress of the "velocity fluctuations" induced

by the periodic thermal perturbation. Davey [1967] extended the theory to finite thermal diffusivity and free surface problems. In several studies [Schubert and Whitehead, 1969, Hinch and Schubert, 1971, Schubert et al., 1971], it was shown that the thermal wave could generated horizontal net flow either in the same or the opposite direction to the thermal wave motion. All these studies have focused on limiting behaviors (small or large Prandtl number/viscosity), and not much attention was given to the maximum net flux, which is often the target of engineering optimizations. In addition, the theory in the referred work are derived from Reynolds averaged equation and developed under the long wave assumption. While these assumptions hold for its astrophysical application (the abnormally high wind speeds in Venus upper atmosphere), it is not so for engineering applications. With those in mind, and considering nowadays computing resources, we feel it is worthwhile to revisit this classic problem and thoroughly explore the parameter space.

In other contexts, Weinert et al. [2008] showed the nonlinearity caused by thermal expansion and temperature-dependent viscosity could also transports fluid. Mao et al. [2013] showed the traveling spatial modulation of surface tension by heat can transport thin liquid films. The net flow is driven by spatial variation of surface tension caused by heating, thus requires an open channel configuration. This could be complicated by three-phase contact line in practical realizations. If the volume expansion by heating is significant, transient heating can be used to transport fluid in the absence of gravity (buoyancy) as demonstrated in another interesting paper by Yariv and Brenner [2004]. Compared with above mechanisms, Boussinesq convection is much simpler in driving fluid motion, with no requirement of free surface, varying viscosity or surface tension. The only essential parts are gravitational field in combination with temperature dependent density causing a variation in buoyancy force. These are the inspiration for us to explore the fluid transport inside an enclosed micro channel using buoyancy driven flows with a traveling wave thermal boundary condition at the bottom wall. In section 2 the mathematical formulation is presented, section 3 presents the asymptotic expansion based on Reynolds number, section

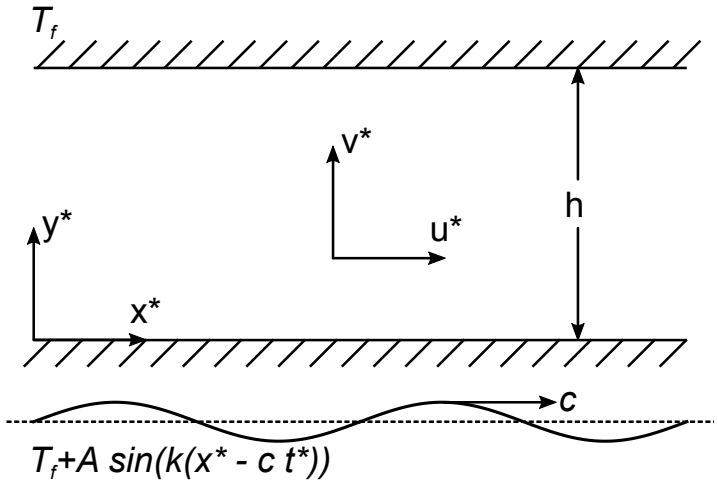

Figure 5.1: A sketch of the 2D microchannel and the thermal boundary conditions.

4 shows typical solution from the analytical and numerical solutions, section 5 presents the parametric study, the optimal parameters for transport is discussed in section 6, section 7 concludes this study with some discussions.

5.2 Mathematical Model

We consider a two dimensional horizontal channel of height h, as illustrated in Figure 1. A Cartesian coordinate system (x^*, y^*) is defined at the bottom wall, with x^* pointing horizontally along the channel and y^* pointing upward. A traveling sinusoidal temperature is imposed at the bottom wall. Since the temperature variation is small compared with the mean temperature, we apply the Boussinesq approximation, and the flow and temperature fields obey

$$\frac{\partial \boldsymbol{u}^*}{\partial t^*} + \boldsymbol{u}^* \cdot \boldsymbol{\nabla}^* \boldsymbol{u}^* = -\frac{1}{\rho}\nabla^* p^* + \nu \nabla^{*2} \boldsymbol{u}^* + \beta g \left(T^* - T_f\right) \boldsymbol{j} \tag{5.1a}$$

$$\nabla^* \cdot \boldsymbol{u}^* = 0 \tag{5.1b}$$

$$\frac{\partial T^*}{\partial t^*} + \boldsymbol{u}^* \cdot \nabla^* T^* = \alpha \nabla^{*2} T^* \tag{5.1c}$$

$$\boldsymbol{u}^*(t^*, x^*, y^* = 0) = \boldsymbol{u}^*(t^*, x^*, y^* = h) = 0 \tag{5.1d}$$

$$T^*(t^*, x^*, y^* = 0) = T_f + A \sin[k(x^* - ct^*)], T^*(t^*, x^*, y^* = h) = T_f \tag{5.1e}$$

Where $\nabla^* = \boldsymbol{i}\partial/\partial x^* + \boldsymbol{j}\partial/\partial y^*$, $\boldsymbol{u}^* = u^*\boldsymbol{i} + v^*\boldsymbol{j}$ is fluid velocity, ρ is density, ν is kinematic viscosity, β is thermal expansion coefficient, g is gravitational acceleration, T^* is temperature, α is thermal diffusivity, T_f is top wall temperature, and A, k and c are, respectively, thermal wave amplitude, wave number, and wave speed. There is no critical Rayleigh number to initiate fluid motion even if $c = 0$, because the non-uniform temperature at the bottom wall leads to uneven buoyancy in the fluid. The problem becomes steady in a reference frame moving with the traveling boundary temperature. We transform the governing equations to this reference frame by defining $x' = x^* - ct^*$, $y' = y^*$, $t' = t^*$, $u' = u^*$, $v' = v^*$, $p' = p^*$, $T' = T^*$. The governing equations and boundary conditions are independent of time in the moving frame:

$$
\begin{aligned}
&(\boldsymbol{u}' - c\boldsymbol{i}) \cdot \boldsymbol{\nabla}'\boldsymbol{u}' = -\frac{1}{\rho}\nabla' p' + \nu\nabla'^2\boldsymbol{u}' + \beta g\,(T' - T_f)\,\boldsymbol{j} \\
&\nabla' \cdot \boldsymbol{u}' = 0 \\
&(\boldsymbol{u}' - c\boldsymbol{i}) \cdot \nabla' T' = \alpha\nabla'^2 T' \\
&\boldsymbol{u}'(x', y' = 0) = \boldsymbol{u}'(x', y' = h) = 0; \\
&T'(x', y' = 0) = T_f + A\sin(kx'), T'(x', y' = h) = T_f
\end{aligned}
\tag{5.2}
$$

Where $\nabla' = \boldsymbol{i}\partial/\partial x' + \boldsymbol{j}\partial/\partial y'$ and $\boldsymbol{u}' = u'\boldsymbol{i} + v'\boldsymbol{j}$. By defining $\boldsymbol{u}' = \boldsymbol{u}^*$ (i.e., we are solving the absolute velocity in the moving reference frame), the boundary conditions for $\boldsymbol{u}$ remain homogeneous, but the momentum and heat equations gain new convective terms. The solution is periodic in x' because of the periodic boundary conditions. There are sufficient differential equations and boundary conditions to solve the problem. The net flow rate along the channel can then be found as

$$
Q' = \int_0^h u'\,\mathrm{d}y' \tag{5.3}
$$

Our main purpose is to find Q'.

5.2.1 Nondimensionalization

The channel height h provides a natural length scale. A dominant balance between the driving buoyancy term and the viscous term in the vertical momentum equation yields a velocity scale $V = \beta g A h^2/\nu$. Based on these scales, a set of dimensionless variables is defined:

$$x = \frac{x'}{h}, y = \frac{y'}{h}, u = \frac{u'}{V}, v = \frac{v'}{V}, p = \frac{hp'}{\rho\nu V}, T = \frac{T' - T_f}{A}, Q = \frac{Q'}{Vh} \tag{5.4}$$

The governing equations and boundary conditions are made dimensionless:

$$\begin{aligned}
&(Re\boldsymbol{u} - Rc\boldsymbol{i}) \cdot \boldsymbol{\nabla u} = -\nabla p + \nabla^2 \boldsymbol{u} + T\boldsymbol{j} \\
&\nabla \cdot \boldsymbol{u} = 0 \\
&(Re\boldsymbol{u} - Rc\boldsymbol{i}) \cdot \nabla T = Pr^{-1}\nabla^2 T \\
&\boldsymbol{u}(x, y = 0) = \boldsymbol{u}(x, y = 1) = 0; \\
&T(x, y = 0) = \sin(Kx), T(x, y = 1) = 0
\end{aligned} \tag{5.5}$$

Where $\nabla = \boldsymbol{i}\partial/\partial x + \boldsymbol{j}\partial/\partial y$, $\boldsymbol{u} = u\boldsymbol{i} + v\boldsymbol{j}$, $Re = Vh/\nu$ is the Reynolds number based on V, $Rc = ch/\nu$ is the Reynolds number based on c, $Pr = \nu/\alpha$ is the Prandtl number, and $K = kh$ is the dimensionless thermal wave number. The definition of the Reynolds number has identical form as the Grashof number, or the Rayleigh number divided by the Prandtl number. The axial flow rate is also made dimensionless:

$$Q = \int_0^1 u \, \mathrm{d}y \tag{5.6}$$

In the next section we seek a solution in the limit of $Re \to 0$ because Re is small in micro channels. A full numerical solution will be presented in section 4.

5.3 Asymptotic Solution

In the limit $Re \to 0$, the dependent variables are expanded in asymptotic series as

$$\begin{aligned} \boldsymbol{u} &= \boldsymbol{u_0} + Re\boldsymbol{u_1} + \cdots \\ p &= p_0 + Rep_1 + \cdots \\ T &= T_0 + ReT_1 + \cdots \end{aligned} \tag{5.7}$$

Where $\boldsymbol{u_0} = u_0\boldsymbol{i} + v_0\boldsymbol{j}$, and $\boldsymbol{u_1} = u_1\boldsymbol{i} + v_1\boldsymbol{j}$. The net horizontal volume flow rate is also expanded as $Q = Q_0 + ReQ_1 + \cdots$.

$$Q_0 = \int_0^1 u_0 \, \mathrm{d}y; \quad Q_1 = \int_0^1 u_1 \, \mathrm{d}y \tag{5.8}$$

The solutions will be presented following the order of expansion.

5.3.1 Leading Order

At the leading order, the governing equations and boundary conditions are

$$\begin{aligned} &-Rc\tfrac{\partial \boldsymbol{u_0}}{\partial x} = -\nabla p_0 + \boldsymbol{\nabla}^2\boldsymbol{u_0} + T_0\boldsymbol{j} \\ &\boldsymbol{\nabla \cdot u_0} = 0 \\ &-Rc\tfrac{\partial T_0}{\partial x} = Pr^{-1}\nabla^2 T_0 \\ &\boldsymbol{u_0}(x,0) = \boldsymbol{u_0}(x,1) = 0; T_0(x,0) = \sin(Kx), T_0(x,1) = 0 \end{aligned} \tag{5.9}$$

The pressure term is then eliminated from the momentum equation by cross differentiation. The horizontal velocity u_0 is eliminated from the combined momentum equation using the

continuity equation, and we are left with

$$\begin{aligned}
&Rc\nabla^2\tfrac{\partial v_0}{\partial x} + \nabla^4 v_0 + \tfrac{\partial^2 T_0}{\partial x^2} = 0 \\
&Rc\tfrac{\partial T_0}{\partial x} + Pr^{-1}\nabla^2 T_0 = 0 \\
&v_0(x,0) = v_0(x,1) = 0; \\
&\tfrac{\partial v_0}{\partial y}(x,0) = \tfrac{\partial v_0}{\partial y}(x,1) = 0; \\
&T_0(x,0) = \sin(Kx), T_0(x,1) = 0
\end{aligned} \tag{5.10}$$

Since the thermal boundary condition is periodic in x, we take

$$\begin{aligned}
v_0 &= \mathrm{Im}[F(y)\mathrm{e}^{\mathrm{i}Kx}] \\
T_0 &= \mathrm{Im}[G(y)\mathrm{e}^{\mathrm{i}Kx}]
\end{aligned} \tag{5.11}$$

Where Im stands for the imaginary part. Substitution of these into 5.10 yields a pair of coupled linear ordinary differential equations:

$$\begin{aligned}
&\frac{\mathrm{d}^4 F}{\mathrm{d}y^4} + \left(\mathrm{i}RcK - 2K^2\right)\frac{d^2 F}{dy^2} - K^2\left(\mathrm{i}RcK - K^2\right)F = K^2 G \\
&\frac{\mathrm{d}^2 G}{\mathrm{d}y^2} + \left(\mathrm{i}RcKPr - K^2\right)G = 0 \\
&F(0) = F(1) = \frac{\mathrm{d}F}{\mathrm{d}y}(0) = \frac{\mathrm{d}F}{\mathrm{d}y}(1) = 0; G(0) = 1, G(1) = 0
\end{aligned} \tag{5.12}$$

The heat equation is solved to give

$$G(y) = \frac{\sin[\sqrt{\mathrm{i}RcKPr - K^2}(1-y)]}{\sin[\sqrt{\mathrm{i}RcKPr - K^2}]} \tag{5.13}$$

With solution for $G(y)$, $F(y)$ is obtained in analytical form using Wolfram Mathematica®, and is listed at the end of this chapter because of its length. The horizontal velocity u_0 is found from v_0 as

$$u_0 = \mathrm{Im}\left(\frac{\mathrm{i}}{K}\frac{\mathrm{d}F}{\mathrm{d}y}\mathrm{e}^{\mathrm{i}Kx}\right) \tag{5.14}$$

Thus, the leading order volume flow rate is found to be zero:

$$Q_0 = \int_0^1 u_0 \, \mathrm{d}y = 0 \tag{5.15}$$

5.3.2 First Order

The first order governing equation and boundary conditions are

$$\begin{aligned}
&\boldsymbol{u_0} \cdot \boldsymbol{\nabla} \boldsymbol{u_0} = -\nabla p_1 + \nabla^2 \boldsymbol{u_1} + T_1 \boldsymbol{j} + Rc \frac{\partial \boldsymbol{u_1}}{\partial x} \\
&\nabla u_1 = 0 \\
&\boldsymbol{u_0} \cdot \nabla T_0 = \frac{1}{Pr} \nabla^2 T_1 + Rc \frac{\partial T_1}{\partial x} \\
&u_1(x,0) = u_1(x,1) = T_1(x,0) = T_1(x,1) = 0
\end{aligned} \tag{5.16}$$

Since the main purpose of this work is to find Q, we only need to focus on the horizontal velocity component u_1, which obeys

$$u_0 \frac{\partial u_0}{\partial x} + v_0 \frac{\partial u_0}{\partial y} = -\frac{\partial p_1}{\partial x} + \frac{\partial^2 u_1}{\partial x^2} + \frac{\partial^2 u_1}{\partial y^2} + Rc \frac{\partial u_1}{\partial x} \tag{5.17}$$

Each of the RHS source terms is periodic (period being half the thermal wave length, or π/K). As the equation is linear, both p_1 and u_1 have period π/K. Averaging equation 5.17 in x direction over the thermal wave length $2\pi/K$ yields

$$\frac{\mathrm{d}\overline{u_0 v_0}}{\mathrm{d}y} = \frac{\mathrm{d}^2 \overline{u}_1}{\mathrm{d}y^2} \tag{5.18}$$

where overline means horizontal average over the thermal wave length. All other terms vanish owing to periodicity in x. Solve equation 5.18 by integrating in y, together with the boundary conditions $\overline{u_1}(0) = \overline{u_1}(1) = 0$ we have

$$\overline{u}_1(y) = \int_0^y \overline{u_0 v_0} \, \mathrm{d}\xi - y \int_0^1 \overline{u_0 v_0} \, \mathrm{d}\xi = \int_0^y \left(\overline{u_0 v_0} - \langle \overline{u_0 v_0} \rangle \right) \, \mathrm{d}\xi \tag{5.19}$$

The angle bracket of the last term denotes vertical average over the entire interval $[0, 1]$, so this term is a constant. The first order volume flow rate Q_1 is found as

$$Q_1 = \int_0^1 \overline{u}_1(y)\,\mathrm{d}y = \int_0^1 \int_0^y (\overline{u_0 v_0} - \langle \overline{u_0 v_0} \rangle)\,\mathrm{d}\xi\,\mathrm{d}y \tag{5.20}$$

Together with Q_0, the volume flow rate Q is

$$Q(Re, Rc, Pr, K) = Q_0 + ReQ_1(Rc, Pr, K) = Re \int_0^1 \int_0^y (\overline{u_0 v_0} - \langle \overline{u_0 v_0} \rangle)\,\mathrm{d}\xi\,\mathrm{d}y \tag{5.21}$$

This concludes the asymptotic solution of the volume flow rate. In the next section we present the numerical method we use to solve Equation (and boundary conditions) 5.5 in their full form, and the characteristic flow pattern revealed by this solution.

5.4 Numerical Method and Typical Solution

We use the finite volume computational fluid dynamics (CFD) toolkit OpenFOAM [Jasak et al., 2007] to solve equation 5.5. Our solver is based on buoyantBoussinesqPimpleFoam solver distributed with OpenFOAM version 3.0. We treat the problem as a transient one and use time marching to reach the steady state. The PIMPLE algorithm, which is a combination of the SIMPLE (semi-implicit method for pressure-linked equations) algorithm and the PISO (pressure-implicit split-operator) algorithm is used to iteratively resolve the incompressibility constraint.To add the extra convective term, the same approach as used by OpenFOAM (and many other CFD code) to handle moving reference frame or dynamic mesh motion is used. The face based flux field is temporarily changed to include the extra convective term, then it is changed back before solving for the pressure constraint. All the customizations of the solver was validated before application to this problem, validation cases include linear convection, pressure distribution in stable Rayleigh-Bénard setup, critical Rayleigh number for the Rayleigh instability (figure 5.2), and several mesh independent studies to validate the second order spatial and temporal accuracy.

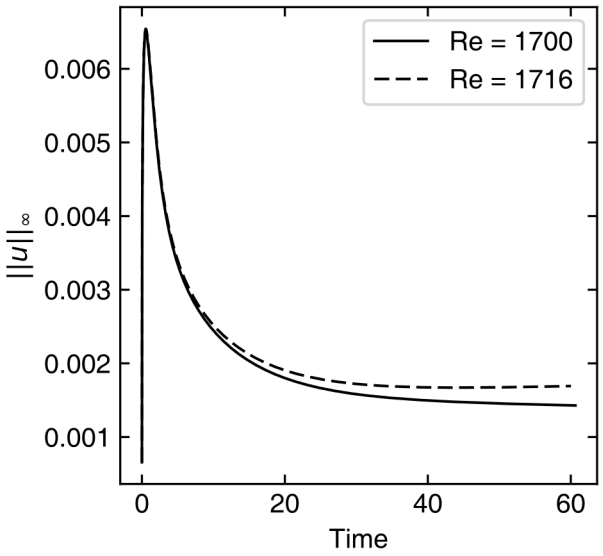

Figure 5.2: Validation of the OpenFOAM solver using the Rayleigh-Bernard instability. The critical Rayleigh number for instability is 1706. We tested two values around this critical value by varying the fluid viscosity. The flow field was given random disturbances at initialization. The disturbances develop in the $Re = 1716$ case while diminish in the $Re = 1700$ case.

Equation 5.5 for $K = 2\pi$ and $Pr = 1$ is solved for five Rc values, each with fifteen different Re values. A 64×128 mesh in the square $[0, 1] \times [0, 1]$ is created. The mesh is uniform in x (64 cells), and has a gentle gradation in y (128 cells). The gradation is introduced such that the mesh better resolves the bottom wall boundary layer. Starting from steady state, the solution is considered converged when the absolute change of the maximum velocity magnitude is less than 10^{-16}, i.e., close to the machine precision. A mesh independence study shows the solutions obtained this way have about three significant digits.

Figure 5.3 shows a typical solution of the temperature and flow field when Re is small. All analytical solutions (solid lines) shown are first order. The flow field is shown through the stream function Ψ. Ψ for the analytical solutions is calculated as follows. After v_0 is obtained with the formula in the appendix, u_0 is calculated numerically with central difference (and some multiplications). With u_0 and v_0, Ψ_0 is integrated with Simpson's

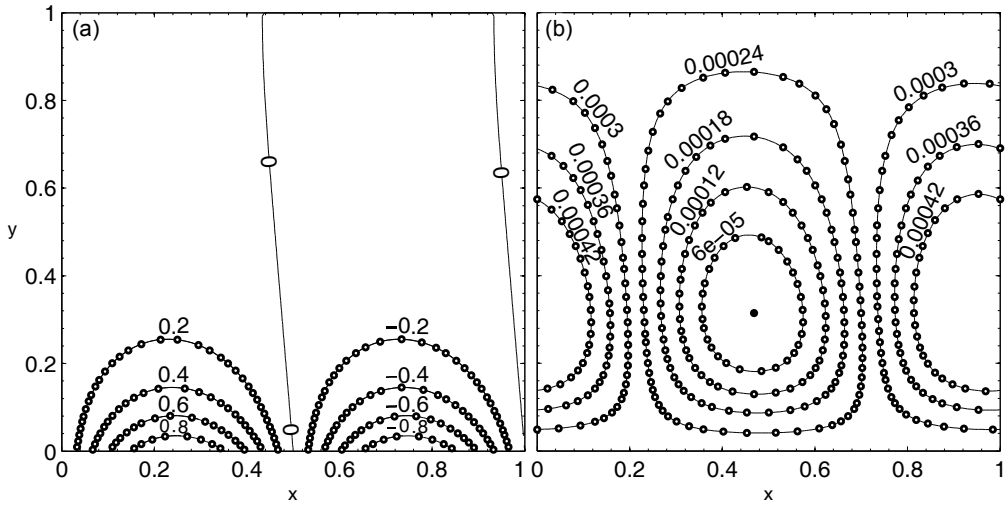

Figure 5.3: A typical solution of temperature T (a) and stream function Ψ (b). Solution parameters are $Rc = Pr = Re = 1$ and $K = 2\pi$. The lines are analytical solutions, and the dots are numerical solutions with OpenFOAM.

rule. The two solutions agrees well with relative error$\sim 0.1\%$. When Re is increased, the first order solution and the numerical solution start to deviate. Figure 5.4 shows such a case for $Re = 1000$. Note that Pr and K are changed as well to explore the analytical solutions' applicability. Regarding the temperature field, the numerical solution displaces slightly upward over the hotter region and downward over the cold region. The first order analytical solutions remains symmetric. The flow pattern undergoes similar changes. The convection cell lifts over the hot region, and sinks over the cold region. Those changes suggest stronger velocity-temperature coupling, and stronger nonlinear effects. The relative error for the case in figure 5.4 is about 2%. It is worth pointer out that the temperature profile in figure 5.3 and 5.4 are similar even though the latter has faster thermal wave speed, due to the latter case's greater thermal diffusivity.

5.5 Effect of Model Parameters

5.5.1 Effect of Re

Equation 5.21 suggests the volume flow rate Q is linearly proportional to the Reynolds number, i.e., the ratio $Q/Re = Q_1 = f(Rc, K, Pr)$ is constant regarding to Re. If higher order terms are included from the asymptotic series, the ratio becomes $Q/Re = Q_1 +$

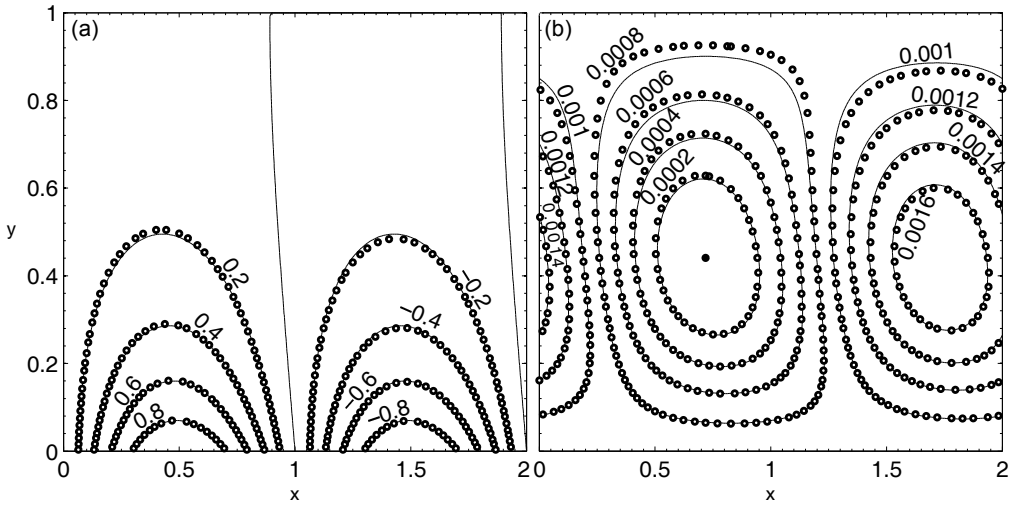

Figure 5.4: A typical solution of temperature T (a) and stream function Ψ (b). Solution parameters are $Rc = 10$, $Pr = 0.1$, $Re = 1000$ and $K = \pi$. The lines are analytical solutions, and the dots are numerical solutions with OpenFOAM. Note the plots are scaled in x direction.

$ReQ_2 + O(Re^2)$. Keeping Rc, K, and Pr fixed, we should eventually be able to observe this deviation by increasing Re. The numerical models discussed in section 5.4 is used to capture the fully non-linearity.

Figure 5.5 summarizes Q/Re from 79 computational results (symbols) and compares them with the analytical solution (lines). As discusses in the last paragraph, the analytical solution predicts a constant ratio independent of Re. The fact that this agrees well with the numerical solution for Re number up to a few thousands is surprising, especially considering that apparent deviations in the velocity field already happened at $Re = 1000$ in figure 5.4. We doubled the mesh resolutions and continued calculating four more $Rc = 10$ cases, using $Re = 2000, 5000, 10000$, and 20000—a further doubling of mesh resolution causes negligible difference in Q. The deviation is revealed, with the trend corresponding to a negative Q_2 value.

The reason that analytical prediction of Q remains valid even when that of the velocity does not can be understood from figure 5.6, which plots the horizontally averaged (over the wavelength $2\pi/K$) velocity profile $\bar{u} = \bar{u}(y)$—the same cases in figure 5.3 and 5.4 are used here. The first order $u_0(y)$ solution consists of simple sin and cos functions and averages

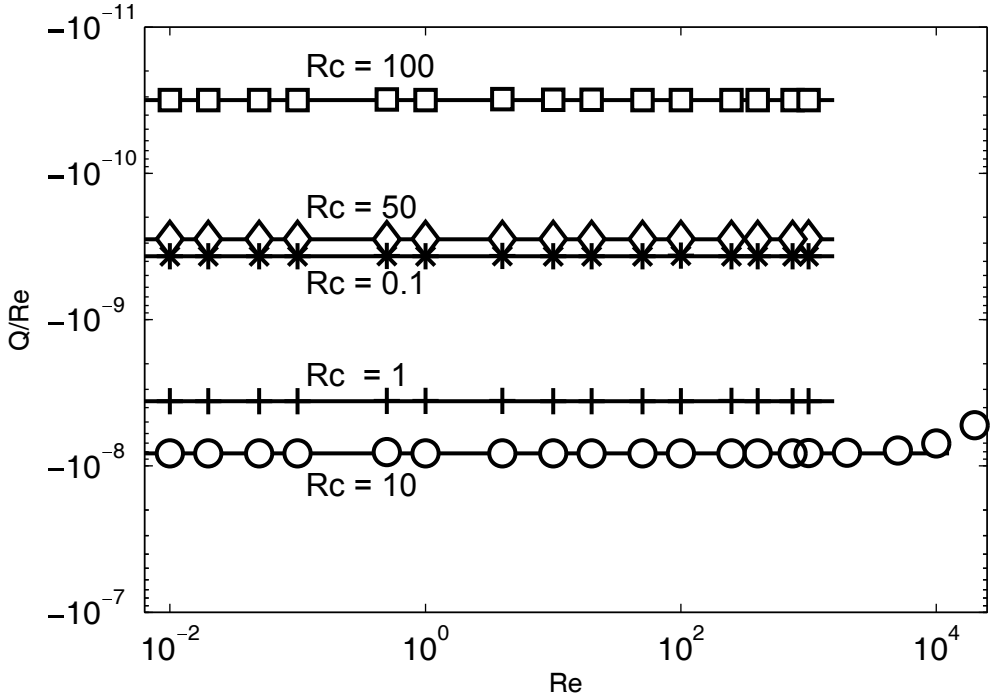

Figure 5.5: Dependence of volume flow rate Q on Re. The x-axis is the Reynolds number, and the y-axis is volume flow rate divided by Reynolds number. Five different Rc values spanning three orders of magnitudes are used, as indicated by the texts. For each Rc, fifteen Re values are used. The range of Re is from 10^{-2} to 10^3, i.e, covers five orders of magnitude. The $Rc = 10$ case has four extra large Re case to show the derivation between numerical and analytical solutions.

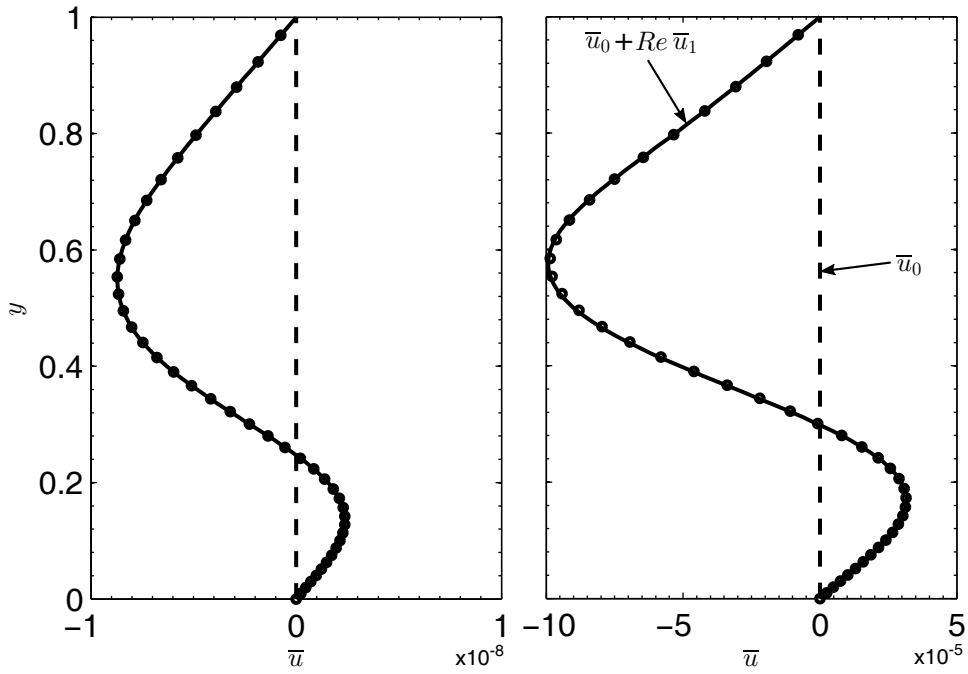

Figure 5.6: The horizontal velocity profile $\bar{u}(y)$. The left and right panels corresponds to the case in figure 5.3 and 5.4, respectively. Shown together in each panel are the numerical solution (dots, every other three data points are shown), the first order solution $\bar{u}_0$ (dashed line), and the second order solution $\bar{u}_0 + Re\bar{u}_1$ (solid line).

to zero $\bar{u}_0(y)$ over the wavelength. The second order u_1 solution contains $\sin^2$ and $\cos^2$ terms and average to nonzero. The agreement between $Re\bar{u}_1(y)$ and the fully nonlinear numerical solution (dots) is very good: the deviations of the velocity field in figure 5.4 mostly cancel each other during the average. Since Q, as the vertical integration of $\bar{u}(y)$, can be considered as an average of average, its range of validity is much greater than $\boldsymbol{u}_0$ solution.

In this chapter we do not worry about the deviations in Q at $Re > 10000$ and restrict ourselves to Re of a few thousands or smaller considering the possible microfluidic application. As the analytical solution matches well with the numerical one, we will rely on it solely to explore the effects of other parameters (K, Rc, and Pr).

5.5.2 Effect of K

Of the four parameters (Re, K, Rc, Pr), K is the first parameter from the thermal wave, the second is Rc. Figure 5.7 shows the dependence $Q(K)$. Qualitatively Q first increases

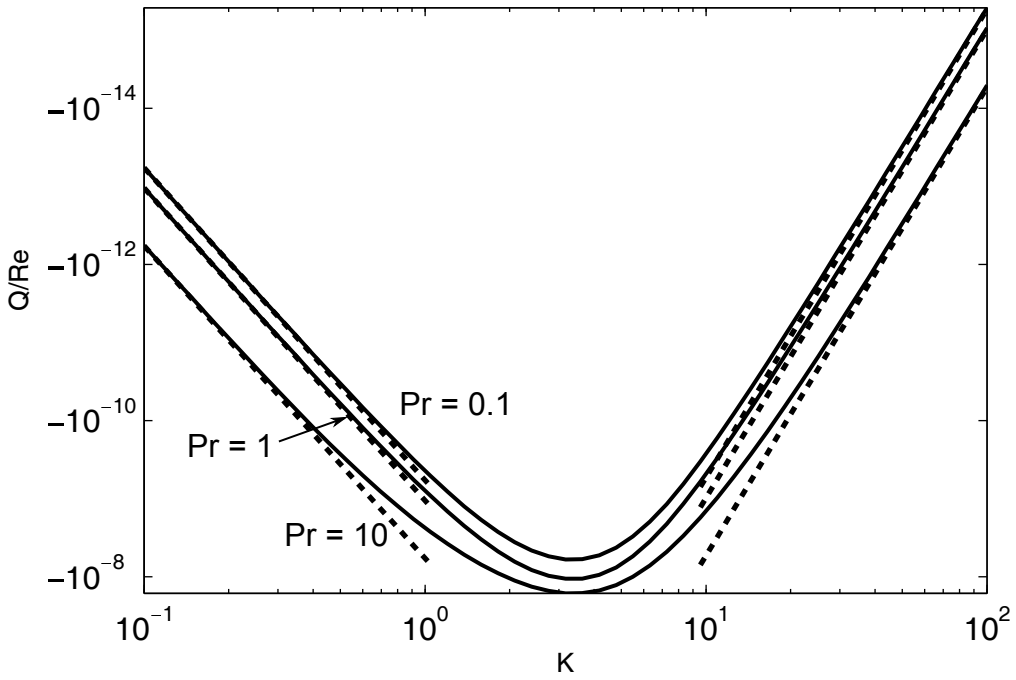

Figure 5.7: Dependence of volume flow rate Q on K, $Rc = 1$, $Re = 1$. Three Pr values are used. The small and large K asymptotic solution are shown as dashed lines.

then decreases as K varies from zero to infinity. The limiting behaviors agree with physical intuition. As $K \to 0$, the thermal wave length goes to infinity. In this limit the thermal boundary condition at the bottom wall is essentially constant. In fact, setting $K = 0$ in $T(x, y = 0) = \sin K(x - ct)$ (boundary condition of equation 5.5) reduces it to constant zero, and both the flow field and volume flow rate become zero. The other limit is $K \to \infty$. Then the alternation of heating and cooling, or the thermal wave frequency is $ck/2\pi$ goes to infinity, and effectively they cancel out. Both the flow field and volume flow rate reduce to zero in this limit. Note that the letting K goes to infinity is quite similar to fixing K but letting Rc goes to infinity as the thermal wave frequency is formed by their product, we will make use of this observation when discussing the effect of Rc.

As all previous studies employed the long wave assumption from the beginning, the asymptotic dependence of Q on K has not been explored and deserves a detailed discussion. We start with the regular $K \to 0$ limit and follow it with the singular $K \to \infty$ limit. Detailed procedures are shown as equation 5.12 is used in its full form.

- $K \to 0$ **limit**

The regular perturbation series is $F = F_0 + KF_1 + K^2F_2 + K^3F_3 + O(K^4)$. Before substituting it into the ODE however, we can estimate the scaling of Q as follows. The ODE for $F(y)$ in equation 5.12 has homogeneous boundary conditions, which means any non-trivial solutions are contributed by the right hand side source. As the source is of order K^2, both F_0 and F_1 are constant zero, or trivial solutions. So F and the vertical velocity have leading order K^2. From the continuity equation the horizontal velocity has order K, and Q has order K^3 from equation 5.21. This conclusion is incorrect however, as will be shown below.

First expand the right hand source term into a power series of K

$$G(y) = (1-y) + \frac{1}{6}\left(2y - 3y^2 + y^3\right) \mathrm{i}RcKPr + O(K^2).$$

Substitute both $G(y)$ and $F(y)$ series into the F equation, expand and collect coefficients of K to have different order ODEs.

$$\begin{aligned}
&{F_0}^{(4)}(y) = 0 \\
&{F_1}^{(4)}(y) + \mathrm{i}RcF_0''(y) = 0 \\
&{F_2}^{(4)}(y) + \mathrm{i}RcF_1''(y) - 2F_0''(y) = 1 - y \\
&{F_3}^{(4)}(y) + \mathrm{i}RcF_2''(y) - 2F_1''(y) - \mathrm{i}RcF_0(y) = \tfrac{1}{6}\left(2y - 3y^2 + y^3\right) \mathrm{i}RcPr \\
&F_i(0) = F_i(1) = F_i'(0) = F_i'(1) = 0 \quad \text{for} \quad i = 0, 1, 2, 3
\end{aligned} \tag{5.22}$$

It is worth mentioning that the long wave equation used in [Stern, 1959, Davey, 1967, Hinch and Schubert, 1971] corresponds to the F_2 equation here with different right hand side according to their specific assumptions. As pointed out previously, $F_0(y) = F_1(y) = 0$. The equations for F_2 and F_3 are solved in sequence, each by simply integrating their ODE

four times and applying the boundary conditions.

$$F_2(y) = \frac{1}{120}\left(3y^2 - 7y^3 + 5y^4 - y^5\right)$$
$$F_3(y) = \frac{\mathrm{i}Rc(-1+y)^2y^2\left(19 + 37y - 50y^2 + 10y^3 + 10Pr\left(11 + 3y - 5y^2 + y^3\right)\right)}{50400}$$

According to the definition of v_0 (equation 5.11) and u_0 (equation 5.14), their product in terms of the perturbation series is $u_0v_0 = \mathrm{Im}[(K^2F_2 + K^3F_3)\mathrm{e}^{\mathrm{i}Kx}] \cdot \mathrm{Im}[\mathrm{i}(KF_2' + K^2F_3')\mathrm{e}^{\mathrm{i}Kx}]$. Expand and taking into account that F_2/F_3 is a real/imaginary only solution, we have

$$\begin{aligned} u_0v_0 = \quad & K^3F_2F_2'\sin(Kx)\cos(Kx) + \\ & K^4\left(F_2'F_{3I}\cos^2(Kx) - F_2F_{3I}'\sin^2(Kx)\right) + \\ & O(K^5), \end{aligned} \tag{5.23}$$

where "I" in the subscript means imaginary part. Average u_0v_0 over the thermal wave length $2\pi\epsilon$, the $O(\epsilon^3)$ term vanishes, and we have

$$\overline{u_0v_0} = \frac{1}{2}K^4\left(F_2'F_{3I} - F_2F_{3I}'\right) + O(K^5) \tag{5.24}$$

This result is interesting in that it involves the product of both the second and third terms. This explains why the earlier estimation gives the wrong $Q \propto K^3$ scaling: it relies on F_2 alone which in fact does not give any net flow. On the other hand, the velocity scaling of it is still correct. Substitute the solutions of $F_2(y)$ and $F_3(y)$ to have $\overline{u_0v_0}$, $Q/Re = Q_1$ is then found through equation 5.20

$$Q_1 = -\frac{1}{2}K^4 \cdot \frac{229(1+Pr)Rc}{217945728000}$$

- $K \to \infty$ **limit**

  This limit leads to a singular perturbation problem and is more complicated. The F equation is reproduced below for easy reference. It can be tempting to start by dropping

lower order terms from equation 5.25, which is incorrect because Rc will then be removed from the equation. This is equivalent to setting $Rc = 0$, which makes the thermal wave stationary and gives no net flow.

$$\begin{aligned} &\frac{\mathrm{d}^4F}{\mathrm{d}y^4} - (2K^2 - \mathrm{i}RcK)\frac{d^2F}{dy^2} + (K^4 - \mathrm{i}RcK^3)F = K^2G(y) \\ &F(0) = F(1) = \frac{\mathrm{d}F}{\mathrm{d}y}(0) = \frac{\mathrm{d}F}{\mathrm{d}y}(1) = 0 \end{aligned} \tag{5.25}$$

Let $1/K = \epsilon$, equation 5.25 becomes

$$\epsilon^4\frac{\mathrm{d}^4F}{\mathrm{d}y^4} - (2\epsilon^2 - \mathrm{i}\epsilon^3Rc)\frac{d^2F}{dy^2} + (1 - \mathrm{i}\epsilon Rc)F = \epsilon^2G(y) \tag{5.26}$$

Let the boundary layer thickness be δ, we get $\delta \sim \epsilon$ by balancing the first and second terms of the left hand side. After rescaling $y/\epsilon = \xi$ we have the inner equation:

$$\frac{\mathrm{d}^4F}{\mathrm{d}\xi^4} - (2 - \mathrm{i}\epsilon Rc)\frac{d^2F}{d\xi^2} + (1 - \mathrm{i}\epsilon Rc)F = \epsilon^2\mathrm{e}^{-\xi}\left(1 + \epsilon\frac{1}{2}\mathrm{i}RcPr\xi\right) \tag{5.27}$$

where the asymptotic expansion

$$\begin{aligned} G(y) &= \frac{\sin[\sqrt{\mathrm{i}RcKPr - K^2}(1-y)]}{\sin[\sqrt{\mathrm{i}RcKPr - K^2}]} \sim \frac{\sinh((K - \mathrm{i}RcPr/2K)(1-y)}{\sinh(K - \mathrm{i}RcPr/2K)} \\ &\sim \frac{\exp((K - \mathrm{i}RcPr/2K)(1-y)}{\exp(K - \mathrm{i}RcPr/2K)} = \exp(-(K - \mathrm{i}RcPr/2K)y) \\ &= \mathrm{e}^{-\xi}\mathrm{e}^{\epsilon\frac{1}{2}\mathrm{i}RcPr\xi} \sim \mathrm{e}^{-\xi}\left(1 + \epsilon\frac{1}{2}\mathrm{i}RcPr\xi\right) \end{aligned} \tag{5.28}$$

is used. There is no source term outside of the boundary layer, and because the top wall boundary condition is homogeneous, the outer solution is zero. Expand F as

$$F_0(\xi) + \epsilon F_1(\xi) + \epsilon^2F_2(\xi) + \epsilon^3F_3(\xi) + O\left(\epsilon^4\right),$$

the equations and boundary conditions for the first few orders are

$$
\begin{aligned}
&F_0{}^{(4)}(\xi) - 2F_0''(\xi) + F_0(\xi) = 0 \\
&F_1{}^{(4)}(\xi) - 2F_1''(\xi) + F_1(\xi) + \mathrm{i}RcF_0''(\xi) - \mathrm{i}RcF_0(\xi) = 0 \\
&F_2{}^{(4)}(\xi) - 2F_2''(\xi) + F_2(\xi) + \mathrm{i}RcF_1''(\xi) - \mathrm{i}RcF_1(\xi) = \mathrm{e}^{-\xi} \\
&F_3{}^{(4)}(\xi) - 2F_3''(\xi) + F_3(\xi) + \mathrm{i}RcF_2''(\xi) - \mathrm{i}RcF_2(\xi) = \tfrac{1}{2}\mathrm{i}RcPr\xi\mathrm{e}^{-\xi} \\
&F_i(0) = F_i(\infty) = F_i'(0) = F_i'(\infty) = 0 \quad \text{for} \quad i = 0, 1, 2, 3
\end{aligned}
\tag{5.29}
$$

Similar to the regular perturbation in the small K limit, the first two terms F_0 and F_1 are constant zero. Substitute them into the equations, we have equation for $F_2(\xi)$ and $F_3(\xi)$

$$
\begin{aligned}
&F_2{}^{(4)}(\xi) - 2F_2''(\xi) + F_2(\xi) = e^{-\xi} \\
&F_3{}^{(4)}(\xi) - 2F_3''(\xi) + F_3(\xi) = \mathrm{i}Rc(F_2(\xi) - F_2''(\xi) + \tfrac{1}{2}Pr\xi\mathrm{e}^{-\xi}) \\
&F_i(0) = F_i(\infty) = F_i'(0) = F_i'(\infty) = 0 \quad \text{for} \quad i = 3, 4
\end{aligned}
\tag{5.30}
$$

According to the definition of v_0 (equation 5.11) and u_0 (equation 5.14), their product is $u_0v_0 = \mathrm{Im}[(\epsilon^2F_2 + \epsilon^3F_3)\mathrm{e}^{\mathrm{i}x/\epsilon}] \cdot \mathrm{Im}[\mathrm{i}(\epsilon^2F_2' + \epsilon^3F_3')\mathrm{e}^{\mathrm{i}x/\epsilon}]$. Expand and again taking into account that F_2/F_3 is a real/imaginary only solution, we have

$$
\begin{aligned}
u_0v_0 = \quad & \epsilon^4F_2F_2'\sin(x/\epsilon)\cos(x/\epsilon) + \\
& \epsilon^5\left(F_2'F_{3I}\cos^2(x/\epsilon) - F_2F_{3I}'\sin^2(x/\epsilon)\right) + \\
& O(\epsilon^6)
\end{aligned}
\tag{5.31}
$$

In averaging u_0v_0 over a thermal wave length $2\pi\epsilon$, the $O(\epsilon^4)$ term vanishes. The remaining terms are

$$
\overline{u_0v_0} = \frac{1}{2}\epsilon^5\left(F_2'F_{3I} - F_2F_{3I}'\right) + O(\epsilon^6) \tag{5.32}
$$

It also involves the product of the second order term and the third term as the small K limit does. Since the boundary layer thickness has order ϵ. We have $Q \propto \epsilon^6 = K^{-6}$. Had we dropped all the Rc terms, we could still get the correct velocity scale, but the flow rate

would be zero, which is incorrect. Solving equation 5.30 analytically:

$$\begin{aligned} F_2(\xi) &= \frac{1}{8}\mathrm{e}^{-\xi}\xi^2 \\ F_3(\xi) &= \frac{1}{96}\mathrm{i}e^{-\xi}\xi^2 Rc(2\xi + 2(\xi+3)Pr + 3) \end{aligned} \tag{5.33}$$

and substitute the solution into equation 5.32, we have

$$\overline{u_0 v_0} = -\frac{1}{768}\epsilon^5 e^{-2\xi}\xi^4(1+Pr)Rc + O(\epsilon^6) \tag{5.34}$$

Since the outer solution is zero, the composite solution is

$$\bar{u}_1(y) = \int_0^y \overline{u_0 v_0}(\tau)\mathrm{d}\tau - c_1 y$$

The constant c_1 is found from the boundary condition $\bar{u}_1(1) = 0$.

$$c_1 = \int_0^1 \overline{u_0 v_0}(y)\mathrm{d}y \approx \frac{(1+Pr)Rc\epsilon^6}{1024}$$

where exponentially smaller terms are dropped. $\bar{u}_1$ is found by substituting in c_1. Integration of $\bar{u}_1$ gives the second order volume flow rate (ϵ replaced with $1/K$)

$$Q_1 = \int_0^1 \bar{u}_1 \mathrm{d}y = -\frac{(1+Pr)Rc}{2048K^6} + O(K^{-7}) \tag{5.35}$$

An interesting observation is the flux is linearly proportional to $(1+Pr)Rc$ in both limits. Note that the large K, or "short wave limit", could refer to a thermal wave of a normal $O(1)$ length scale that travels beneath an infinitely deep channel. In this limit the flux is directly proportional to how fast the thermal wave moves. The second interesting observation is that the relation $Q \propto (Pr+1)$, which was found previously by Davey [1967], holds for both long and short wave limits. A key implication of this proportionality is a net flow rate exists even with instantaneous thermal response ($Pr = 0$), as first demonstrated by Stern

[1959]. The asymptotic solutions in both limits are shown in figure 5.7. They are in good agreement with the solution of the full ODE in their respective limits.

5.5.3 Effect of Rc

Figure 5.5 already shows the nontrivial dependence of Q on Rc: the flux of $Rc = 0.1$ and $Rc = 50$ cases are quite close, and both are smaller (in terms of absolute value) than the flux of $Rc = 10$. The dependence of Q on Rc is well studied. Below we convert the diverse dimensionless groups used in the referred work all to ours. Stern [1959] prescribed a traveling *volumetric*—instead of boundary—temperature field and obtained that

$$Q_1 = \begin{cases} -5 \times 10^{-7} RcK^4 & KRc \quad \text{small} \\ \dfrac{1}{48\sqrt{2}} Rc^{-5/2} K^{-1/2} & KRc \quad \text{large} \end{cases}$$

This second result suggests that the net flow rate is in the same direction as the thermal wave for large KRc. In the paper it was estimated that the reversal of flux direction happens when $KRc \sim 10^2$. However this reversal was not supported by later studies Davey [1967] or our simulation. The second result is likely caused by the artificial assumption of the volumetric temperature field. Davey [1967] improved upon this and used thermal boundary conditions at *both* walls. The heat equation without convection effect is used so the temperature field itself becomes an unknown to found. The volume flow rate was found to be

$$Q_1 = \begin{cases} -\dfrac{1+Pr}{12!} RcK^4 & KPrRc \quad \text{small} \\ -\dfrac{1}{4Pr^2(1+Pr)} Rc^{-4} K^{-1} & KPrRc \quad \text{large} \end{cases}$$

The small Rc limit is similar to that of Stern [1959], with a correction of the numerical coefficient. The large Rc limit however is opposite to thermal wave direction, and it is proportional to Rc^{-4} instead of $Rc^{-5/2}$. This means the assumptions made in the governing equation can directly affect the limiting asymptotic behavior. We plotted the scaling by Davey [1967] in figure 5.8 as solid segment, both capture the limiting behaviors well.

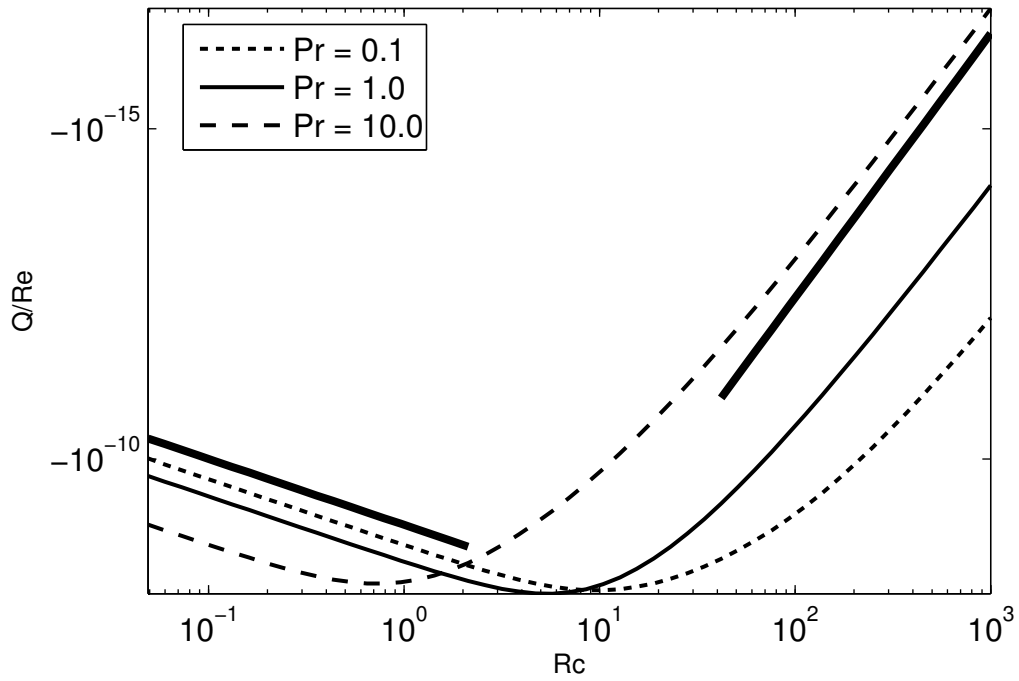

Figure 5.8: Dependence of volume flow rate Q on Rc, with $Rc = 1$ and $K = 2\pi$. Three Prandtl numbers Pr that spanning three orders of magnitudes are used: 0.1, 1.0, and 10.0. The thick line segment are the scaling Rc^1 and Rc^{-4} from equation 5.5.3, an arbitrarily selected numerical coefficient is used.

Like its dependence on K, Q first increases then decreases with Rc, and it vanishes in the two limits. The limiting behavior is easily understood. When $Rc = 0$, the flow consists of closed convection cells and there is no net axial flow rate ($Q = 0$). It should be noted though that the velocity field is nonzero. When $Rc \to \infty$, the thermal wave is traveling at high speed. The heating and cooling at the bottom wall alternate quickly. Their effects have little time to fully develop and tend to cancel each other. Unlike the $Rc = 0$ limit, here both the convection and the flow rate Q vanish. The Rc value that maximizes Q, i.e., optimal for fluid transport purpose is sensitive to the Prandtl number, which spans two orders of magnitude. The magnitude of Q, on the other hand, changes little with Pr. We will discuss the optimal transport parameter in section 5.6.

5.5.4 Effect of Pr

Figure 5.9 shows the dependence of Q on Pr, note that real world fluid's range of Prandtl number is much smaller (0.1–10) compared with what was shown. The limiting behaviors of $Q(Pr \to 0)$ is different from the small K or small Rc limits, in that Q approaches a non-zero constant. Of course the limit $Pr = 0$ means infinitely large thermal diffusivity, or instant thermal diffusion across the channel height. This limit has been discussed several times already. On the other hand, as Pr increases, either by increasing viscosity or decreasing thermal diffusivity, Q eventually vanishes for apparent reasons.

The scaling with small Pr limit can be easily obtained by reusing previous result by noting that Pr only appears together with other parameters as $KPrRc$. This was already reflected in figure 5.7 where the error in the asymptotic solutions decreases as Pr decreases, for the reason that smaller Pr means smaller higher order terms neglected in the $G(y)$ expansion. In the previous two subsections both small K or small Rc has the scaling with $Q_1 \propto (1 + Pr)$. This scaling fits the ODE solution very well by adjusting its numerical coefficient. Pr differs from other parameters in that it is a fluid property, so it is not subject to flexible changes. For these reasons we will not explore Pr in greater detail here.

Just like figure 5.7 and 5.8, there is an optimal Pr value for fluid transport. We will continue studying Pr in the next section where the combination of Pr, K and Rc that maximize Q is discussed in detail.

5.6 Optimal Parameter for Transport

By the section title we mean the parameter combinations that maximize the amplitude of Q. Re can be taken away since its effect is straightforward, and we are left with three parameters (Rc, Pr, K). The following method is used to calculate the desired optimal parameter combinations. First a list of (Pr, K) pairs are generated. The values of Pr sample a prescribed range (here 0.1–10), so are the values of K (0.32–50.12). For each (Pr, K) pair, we calculate a list of (Rc, Q) pairs, then the Rc value where Q_i is the maximum is found by local curve fitting: identify the index i where Q is the maximum; then fit a

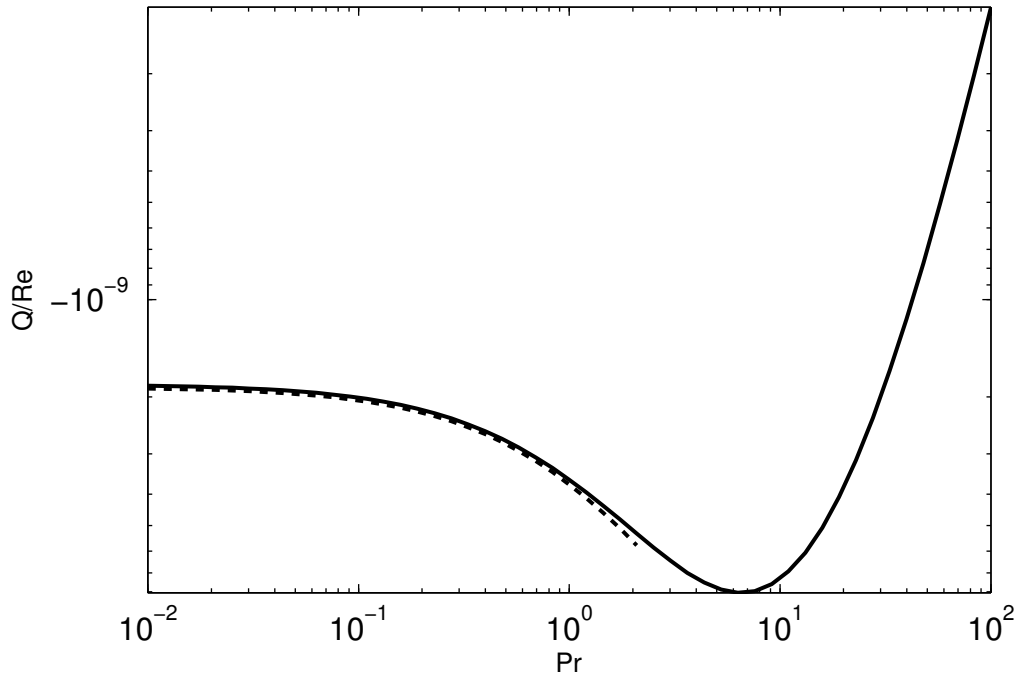

Figure 5.9: Dependence of volume flow rate Q on Pr. Other parameters have fixed values: $Rc = 1$, $Re = 1$, $K = 2\pi$. The range of Pr is purposely chosen to show the entire trend. The dashed line shows the scaling $1 + Pr$, the numerical coefficient is chosen to better fit the solid curve.

parabola through three points: (Rc_{i-1}, Q_{i-1}), (Rc_i, Q_i), and (Rc_{i+1}, Q_{i+1}). Suppose the fitting parabola is $p_1x^2 + p_2x + p_3$, then the optimal Rc is $-p_2/(2p_1)$, and the optimal Q is $-p_2/(4p_1) + p_3$. The results are summarized in figure 5.10.

From the figure, Pr has little effects on the optimal K, but significantly affects the optimal Rc. In addition, Pr does not have much effect on the magnitude of Q (within an order of magnitude). All these are consistent with figure 5.7 and 5.8 and the accompanying discussions. The most interesting character however, is the appearance of a surface folding onto itself. Regardless of thermal wave speed (Rc) or fluid property (Pr), the maximum flow rate occurs in a narrow interval of K, within $[2, 5]$ by visual inspection. The fold appears to change from sharp to a smoother one as Pr decreases. This is only a visual artifact as shown by the upper left inset, where a zoom in view of the fold of the leftmost $Pr = 10$ curve is shown. However, the curvature is indeed much greater compared with, say, that along the $Pr = 1$ curve. The surface not only folds in K, but also in Pr. This is

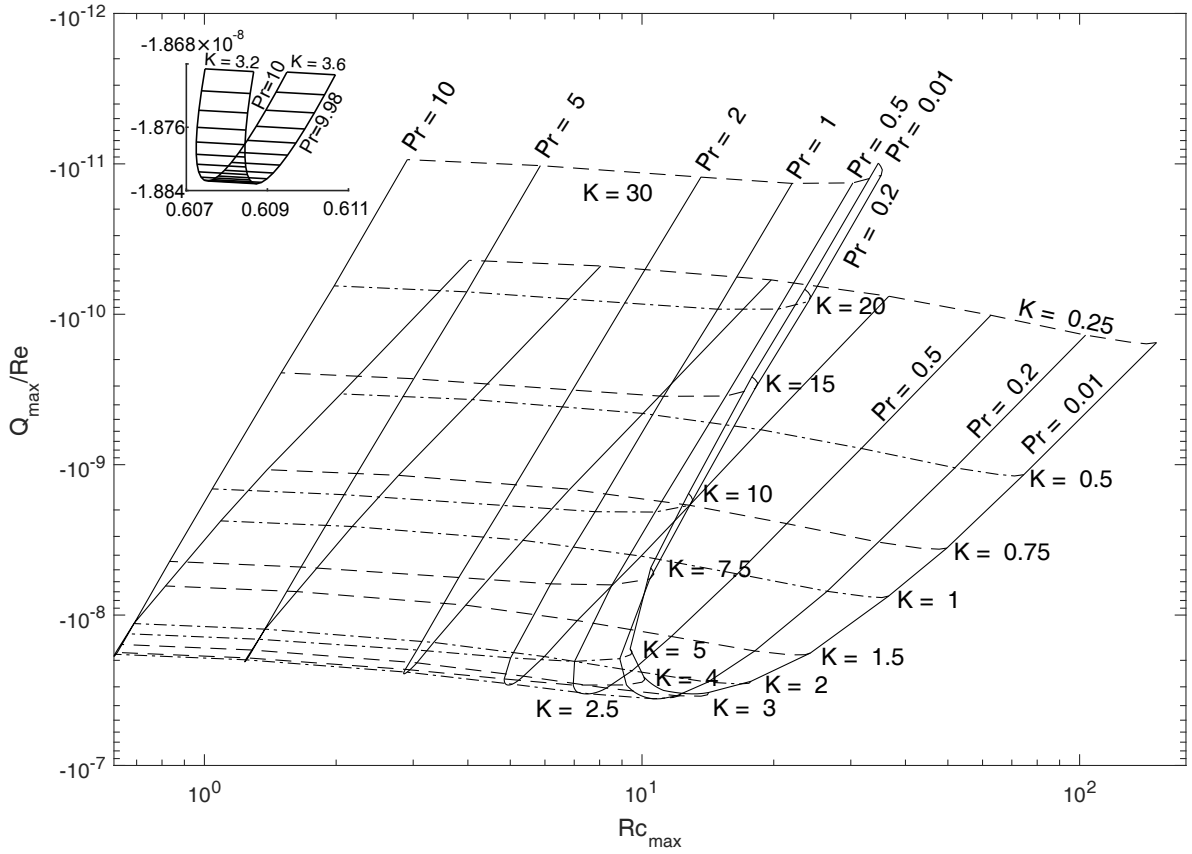

Figure 5.10: Optical volume flow rate Q_{max} versus the optimal Rc_{max} value, for different combinations of Pr and K. Constant Pr curves are drawn in solid, while constant K curves are drawn in dash with alternating dash patterns. The inset is a zoom in to the "turning point" of the $Pr = 10$ curve (a zoom-in box is too small to draw on the original figure).

visible around the crossing of $Pr = 0.01$ and $K = 30$. We won't discuss it in detail because of the unpractical values of Pr.

Figure 5.10 shows that the optimal Rc value decreases with Pr, and it first decreases then increases with K. A heuristic relation between those parameters can be derived as follows. The system has two different time scales in dimensional values: the thermal wave time scale λ/c, and the thermal diffusivity time scale h^2/α. Balancing the two gives $c_{max} = \alpha\lambda/h^2$. Converting to dimensionless parameters defined by us, we have $KPrRc_{max} = 2\pi$. This scaling works if the thermal boundary layer thickness is of order h, i.e., the thermal effect penetrates the entire channel height. When K is large, the vertical extension of the thermal effect is limited. The top wall appears as if it locates at $+\infty$, so h is no longer the proper scale of the thermal boundary layer. The only length scale in the system is the thermal wave length λ. Use it to replace h, we could derive another scaling relation $K^{-1}PrRc_{max} = (2\pi)^{-1}$. The argument here can be loosely compared with the driven damped harmonic oscillator in classical mechanics: the largest oscillation amplitude occurs when the system's natural frequency matches the external forcing's frequency.

Figure 5.11 shows both scalings are correct in their respective K limit. Following this line of argument, the turning point of the curves in figure 5.11 corresponds to where the two scalings give identical results. This happens when $K = 2\pi$, or in terms of dimensional quantities, when the channel height h equals the thermal wave length λ. This estimation is within the same order of magnitude as the values given by numerical solutions. We may also reuse the large and small K asymptotic solutions, equal the two gives another estimation of $K_{max} \approx 4.0$.

5.7 Discussion and Conclusion

5.7.1 Flow Pattern

We have shown that nonlinearity is the cause of the horizontal net flow, and derived its approximation that is valid for Re up to $O(10^3)$. The approximation however is in terms of horizontally averaged velocity and does not capture the flow field. Thus it can not answer

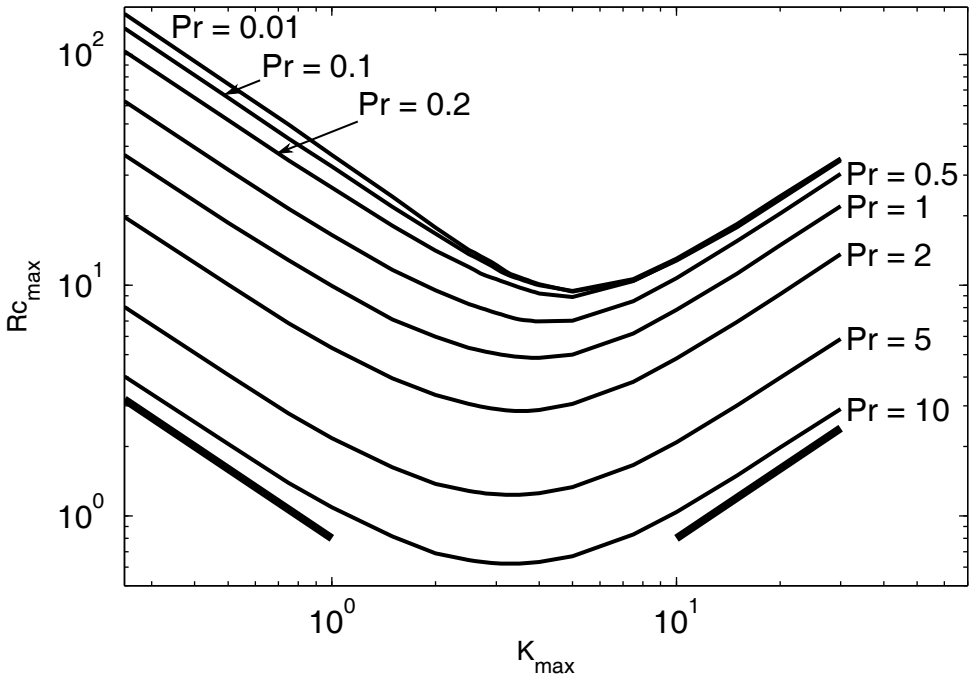

Figure 5.11: Dependence of optimal Rc_{max} on K. $Re = 1$ for all cases. The same data as in figure 5.10 is used. The left and right thick segments represent K^{-1} and K^1 scaling respectively.

interesting questions like what's the qualitative change when Rc deviates from 0 and starts generating the net flux. For this reason we plot the flow field of a high Reynolds number case ($Re = 10000, Rc = 10, Pr = 1, K = 2\pi$) in figure 5.12. The flow rate of this case was shown as a data point in figure 5.5, where it has noticeable deviation from analytical result. The flow is characterized by distorted convective cells and, more importantly, a winding stream in between the cells. The separating streamlines were identified by trial and error (try with different Ψ values and observe the contour. More accurate methods could be used, but this simple method can quickly narrow down the value range and give four significant digits Ψ value for the two separating streamlines).

The winding stream is responsible for the net horizontal flow rate Q. This is shown by calculating the flow rate across different sections of the flow field. The section ends start either from the separating streamline (by linear interpolation) or from the walls. The flow rate across section A, D, and E are all close to the Q value; while those across the convection cells (B and C) are several orders of magnitude smaller—the nonzero values are

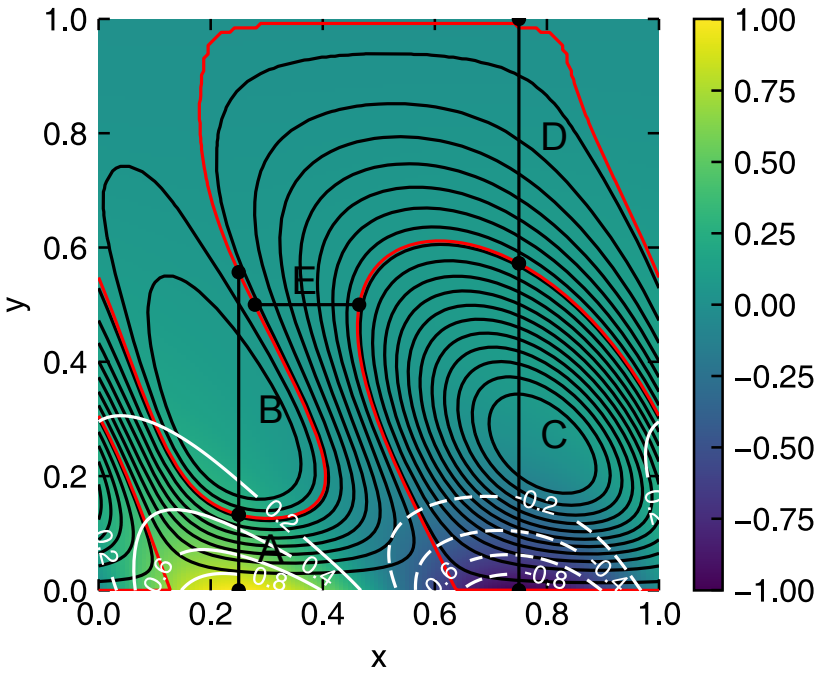

Figure 5.12: Temperature and flow pattern for $Rc = 10$, $Pr = 1$, $K = 2\pi$, and $Re = 10000$. Both the white numbered contours and the color plot shows the temperature field. The solid black curves are the streamlines. The two separating streamlines are colored in red. Five cross sections (marked by capital letters) are used to calculate flow rate across them, which are $Q_A = -7.029 \times 10^{-5}, Q_B = -3.647 \times 10^{-8}, Q_C = -9.610 \times 10^{-8}, Q_D = -7.032 \times 10^{-5}$, and $Q_E = -6.988 \times 10^{-5}$, while the horizontal average flow rate is $Q = -7.032 \times 10^{-5}$.

likely caused by numerical errors, and more accurate calculation should show they are zero to machine precision. It is worth mentioning that he winding stream is narrower but still observable for $Re = 2000$ (other parameters the same), but it is hardly noticeable when $Re = 1000$ or less. We chose the $Re = 10000$ case to plot only because all the interesting features are more easily distinguishable.

5.7.2 Energy and Force Balance

We check the energy balance of the system by integrating the governing equations over the domain. For temperature equation of equation 5.5, its left hand side is first transformed

into the conservative form:

$$(Re\boldsymbol{u} - Rc\boldsymbol{i}) \cdot \nabla T = \nabla \cdot ((Re\boldsymbol{u} - Rc\boldsymbol{i})T) - (\nabla \cdot (Re\boldsymbol{u} - Rc\boldsymbol{i}))T$$

The second terms is zero. Apply the divergence theorem, the resultant surface integration $\oint((Re\boldsymbol{u} - Rc\boldsymbol{i})T)d\boldsymbol{S}$ over the domain boundary vanishes due to the non-slip boundary conditions and the wall direction being perpendicular to $\boldsymbol{i}$. Then integration of the right hand side equals zero as well, or

$$-\oint \nabla T d\boldsymbol{S} = 0$$

This means heat energy is conserved and is not drained for driving the fluid motion. To identify the energy source of the flow, multiply the momentum equation with $\boldsymbol{u}$ and integrate over the domain. After making use of the divergence theorem and wall boundary conditions, the same equation as in Davey [1967] can be derived.

$$2\int_V e_{ij} : e_{ij} = \int_V Tv dv$$

v is the *vertical* velocity component. This shows the viscous dissipation of the flow field is balanced by the "thermal work" instead of "thermal energy". We calculated the above integrations for the case shown in figure 5.12 numerically. The total viscous dissipation and thermal work differ by less than 0.1%. Similarly, the heat flux at the top and bottom walls cancel each other, the relative difference of their numerical values is about 1×10^{-6}. The heat flux integration is more accurate because surface integrations conform with the underlying finite volume view of the numerical method. This interesting energetics was caused by the Boussinessq approximation and was discussed in detail by Mihaljan [1962]. There it was demonstrated that "Boussinesq energy transformations and entropy production depend on the higher-order systems because of the peculiar simplification of the thermodynamic energy equation". In other words, to capture the transformation between internal and

kinetic energy would require including the compressible effect which was neglected during Boussinesq approximation.

Another peculiar result is the flow experiences no *net* wall shear stress. This is apparent for the zeroth order velocity solution, which averages to zero in the horizontal direction. For the first order solution, take the y-derivative of equation 5.19 and use the nonslip wall boundary condition,

$$\frac{\mathrm{d}\overline{u}_1(y)}{\mathrm{d}y} = \overline{u_0 v_0}(y) - \langle \overline{u_0 v_0} \rangle = -\langle \overline{u_0 v_0} \rangle \quad \text{at the wall.}$$

As the top and bottom walls have opposite normal directions, their wall shear stresses cancel. The zero net stress was mention in passing by Stern [1959] for his $\bar{u}_1$ solution. In Hinch and Schubert [1971], the averaged velocity profile was found to exert zero stress at both walls using the simplified long wave equations. However, it can be shown that not just the first two order solutions, but the fully nonlinear flow experiences no net stress from the walls. Starting from the governing dimensional momentum equation, which is reproduced below (asterisk dropped):

$$\frac{\partial \boldsymbol{u}}{\partial t} + \boldsymbol{u} \cdot \boldsymbol{\nabla} \boldsymbol{u} = -\frac{1}{\rho} \nabla p + \nu \boldsymbol{\nabla}^2 \boldsymbol{u} + \beta g \left(T - T_f \right) \boldsymbol{j}$$

Do a volume integration over the domain, the time derivate vanishes at steady state. The convection term becomes

$$\int_V \boldsymbol{\nabla} \cdot \boldsymbol{u}\boldsymbol{u} dv = \oint \boldsymbol{u}\boldsymbol{u} d\boldsymbol{S} = 0$$

and vanishes due to nonslip and periodic boundary conditions. The pressure terms becomes:

$$\frac{1}{\rho} \int_V \nabla p \, dv = \frac{1}{\rho} \oint p \, d\boldsymbol{S} = I_0 \boldsymbol{j}$$

the pressure does not contribute to net horizontal forcing due to periodic boundary condition, but it may contribute to a net vertical force in the j direction. The Boussinesq forcing

term integrates to

$$\beta g \boldsymbol{j} \left(\int_V T dv - \int_V T_f dv \right) = I_1 \boldsymbol{j}$$

so the net force is also in the vertical direction. The last term is the viscous term.

$$\nu \int_V \boldsymbol{\nabla}^2 \boldsymbol{u} dv = \nu \oint_S \boldsymbol{\nabla u} \cdot d\boldsymbol{S} = 0\boldsymbol{i} + I_2 \boldsymbol{j}$$

The last equality is from the observation that there is not other terms in the momentum equation that gives horizontal forcing. Another conclusion is the vertical force balance is between the volumetric temperature forcing and the pressure at the wall. Still using the case in figure 5.12, the total shear stress on each of the two walls differs relatively by 1×10^{-6}. Note that a similar conclusion can be made for the temperature field: that there is no net heat flux in or out. The horizontally averaged temperature profile $\bar{T}(y)$ can be derived similar to that of $\bar{u}(y)$:

$$\bar{T} = Re\bar{T}_1(y) = RePr \int_0^y \left(\overline{T_0 v_0} - \langle \overline{T_0 v_0} \rangle \right) \, \mathrm{d}\xi \tag{5.36}$$

The rest of the argument is similar to that for $\bar{u}$ and omitted. Two typical temperature profiles corresponding to the cases in figure 5.3 and 5.4 are shown in figure 5.13 for completeness.

5.7.3 Conclusion

So far we have been discussing using dimensionless values. To connect with physical world, it is interesting to see the dimensional form of the flux. The scaling of Q is $ReVh$. Expand Re and V, we have $Q \sim \beta^2 g^2 A^2 h^6 \nu^{-3}$. The flux is very sensitive to the channel width, and the fluid viscosity. Take $\beta = 1.0 \times 10^3 K^{-1}$, $g = 10m \cdot s^{-2}$, $A = 10K$, $h = 1mm$, and $\nu = 10^{-6} m^2 \cdot s^{-1}$, we have the vertical velocity scale $V = 0.1m \cdot s^{-1}$, and the Reynolds number $Re = 100$. If further assume the thermal wave number is close to π, the thermal wave speed is about $1cm \cdot s^{-1}$ ($Rc = 10$), and the fluid Prandlt number to be 0.2, we have

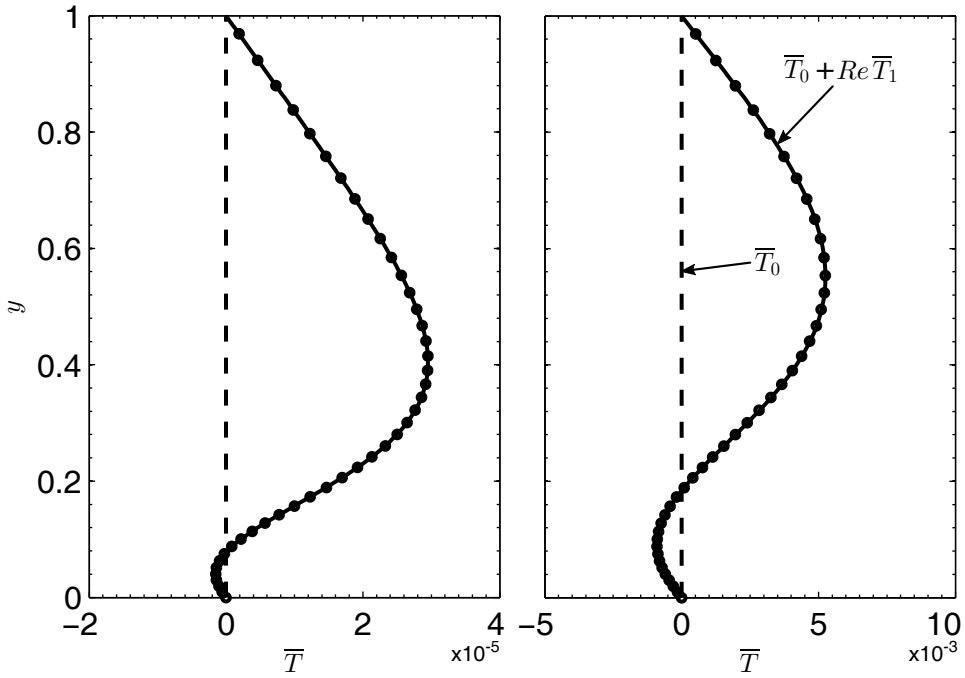

Figure 5.13: The horizontally averaged temperature profile $\bar{T}(y)$. The left and right panels corresponds to the case in figure 5.3 and 5.4, respectively. Shown together in each panel are the numerical solution (dots, every other three data points are shown), the first order solution $\bar{T}_0$ (dashed line), and the second order solution $\bar{T}_0 + Re\bar{T}_1$ (solid line).

the largest possible dimensionless axial flow rate $Q/Re \sim 3 \times 10^{-8}$, or $3 \times 10^{-8} m^2 \cdot s^{-1}$ with dimensions. The corresponding vertically and horizontally averaged speed is $0.03 mm \cdot s^{-1}$.

5.8 Solutions

Define

$$2a = \mathrm{i}RcK - 2K^2, b = -K^2(\mathrm{i}RcK - K^2), l = K^2, d = \sqrt{\mathrm{i}RcKPr - K^2} \tag{5.37}$$

$$s = \sqrt{a^2 - b}, \quad p = \sqrt{-a - s}, q = \sqrt{-a + s} \tag{5.38}$$

$$D = 2\left(b - 2ad^2 + d^4\right)\left(a\left(-1 + e^{2\alpha}\right)\left(-1 + e^{2\beta}\right) + \left(1 + e^{2p} + e^{2\beta} - 4e^{p+\beta} + e^{2(p+q)}\right)\alpha\beta\right)$$

The analytical solution:

$$F(y) = l/D(C_1 e^{yp} + C_2 e^{-yp} + C_3 e^{yq} + C_4 e^{-yq}) + \frac{l \csc(d) \sin(d - dy)}{b - 2ad^2 + d^4} \qquad (5.39)$$

The coefficients are:

$$\begin{aligned} C_1 &= ae^{2q} - pe^{2q}q + 2pqe^{p+q} - pq - a + d\left(-2qe^{p+q} + e^{2q}(p+q) - p + q\right)\cot(d) + \\ &\quad d\left(e^{p+2q}(q-p) + e^{p}(p+q) - 2e^{q}q\right)\csc(d) - e^{2q}s + s \end{aligned}$$

$$\begin{aligned} C_2 &= -e^{p}\left(ae^{p+2q} - e^{p}a + e^{p}pq - 2pe^{q}q + pqe^{p+2q} + \right. \\ &\quad d\left(e^{p+2q}(q-p) + e^{p}(p+q) - 2e^{q}q\right)\cot(d) + d\left(-2qe^{p+q} + e^{2q}(p+q) - p + q\right)\csc(d) \\ &\quad \left. -se^{p+2q} + e^{p}s\right) \end{aligned}$$

$$\begin{aligned} C_3 &= e^{2p}a - e^{2p}pq + 2pqe^{p+q} - pq - a - d\left(2pe^{p+q} - e^{2p}(p+q) - p + q\right)\cot(d) \\ &\quad + d\left(e^{2p+q}(p-q) + e^{q}(p+q) - 2e^{p}p\right)\csc(d) + e^{2p}s - s \end{aligned}$$

$$\begin{aligned} C_4 &= -e^{q}\left(ae^{2p+q} + a\left(-e^{q}\right) - 2e^{p}pq + pe^{q}q + pqe^{2p+q} + \right. \\ &\quad + d\left(e^{2p+q}(p-q) + e^{q}(p+q) - 2e^{p}p\right)\cot(d) - d\left(2pe^{p+q} - e^{2p}(p+q) - p + q\right)\csc(d) \\ &\quad \left. + se^{2p+q} - e^{q}s\right) \end{aligned}$$

For $Pr = 1$, the solution has a simpler form. Define $a = \sqrt{iRcK - K^2}$, $b = K$, the solution is

$$\begin{aligned} F(y) &= 1/\mathcal{D}(\mathcal{C}_1 \cos(ay) + \mathcal{C}_2 \sin(ay) + \mathcal{C}_3 e^{-by} + \mathcal{C}_4 e^{by}) \\ &\quad - \frac{\csc(a)\left(2a\left(a^2+b^2\right)y\cos(a-ay) + \left(5a^2+b^2\right)\sin(a-ay)\right)}{4(a/b)^2\left(a^2+b^2\right)^2} \end{aligned}$$

In which

$$\mathcal{D} = 4(a/b)^2 \left(a^2+b^2\right)^2 \left((a-b)(a+b)\sinh(b) + 2ab\csc(a)(\cos(a)\cosh(b)-1)\right)$$

$$\begin{aligned}\mathcal{C}_1 &= 2a^2 b\csc(a)\left(-\left(a^2+b^2\right)\cot(a) + \csc(a)\cosh(b)\left(a^2+2a\sin(2a)+b^2\right) - 4a\right) \\ &\quad + \left(3a^4 - 6a^2b^2 - b^4\right)\sinh(b)\end{aligned}$$

$$\begin{aligned}\mathcal{C}_2 &= \csc(a)\left(8a^3 b\sin(a)\cosh(b) - 2ab\left(a\left(a^2+b^2\right) + \left(b^2-3a^2\right)\cot(a)\right)\right. \\ &\quad \left. -2ab\csc(a)\left(b\left(a^2+b^2\right)\sinh(b) + \left(3a^2-b^2\right)\cosh(b)\right) + \left(-3a^4+6a^2b^2+b^4\right)\cos(a)\sinh(b)\right)\end{aligned}$$

$$\mathcal{C}_3 = \frac{1}{2}a\left(a^2+b^2\right)\csc(a)\left(-2\left(a^2+b\right) + 2ab\cot(a) + e^b\csc(a)(-2ab + b\sin(2a) + a - a\cos(2a))\right)$$

$$\mathcal{C}_4 = ae^{-b}\left(a^2+b^2\right)\left(e^b\left(a^2-b\right)\csc(a) - ab\csc^2(a) + b\cot(a)\left(ae^b\csc(a)+1\right) - a\right)$$

References

M. Abramowitz and I. A. Stegun. *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables*. 1964. ISBN 0486612724.

M. Agrawal, S. Ookawara, and K. Ogawa. Drag force formulation in macroscopic particle model and its validation. 2009.

M. A. Ambroso, R. D. Kamien, and D. J. Durian. Dynamics of shallow impact cratering. *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics*, 72(4), 2005. ISSN 15393755.

B. Andreotti, Y. Forterre, and O. Pouliquen. *Granular media: between fluid and solid*. Cambridge University Press, 2013.

G. K. Batchelor. An Introduction to Fluid Dynamics, 1967. ISSN 00319228.

B. Bigot, T. Bonometti, L. Lacaze, and O. Thual. Computers & Fluids A simple immersed-boundary method for solid fluid interaction in constant- and stratified-density flows. *COMPUTERS AND FLUIDS*, 97:126–142, 2014. ISSN 0045-7930.

C. Brennen, K. Sieck, and J. Paslaski. Hydraulic jumps in granular material flow. *Powder Technology*, 35(1):31–37, 1983.

N. V. Brilliantov and T. Pöschel. *Kinetic theory of granular gases*. Oxford University Press, 2010.

C. S. Campbell. Granular material flows–an overview. *Powder Technology*, 162(3):208–229, 2006.

M. Cates, J. Wittmer, J.-P. Bouchaud, and P. Claudin. Jamming, force chains, and fragile matter. *Physical review letters*, 81(9):1841, 1998.

P. Cundall and O. Strack. A discrete numerical model for granular assemblies. *Geotechnique*, 331(May):331–336, 1979.

A. Davey. The motion of a fluid due to a moving source of heat at the boundary. *Journal of Fluid Mechanics*, 29(01):137–150, 1967.

J. Duran. *Sands, powders, and grains: an introduction to the physics of granular materials*. Springer Science & Business Media, 2012.

N. T. M. Eldabe, M. F. El-Sayed, A. Y. Ghaly, and H. M. Sayed. Mixed convective heat and mass transfer in a non-Newtonian fluid at a peristaltic surface with temperature-dependent viscosity. *Archive of Applied Mechanics*, 78(8):599–624, oct 2007. ISSN 0939-1533.

A. L. Fogelson and C. S. Peskin. A fast numerical method for solving the three-dimensional stokes' equations in the presence of suspended particles. *Journal of Computational Physics*, 79(1):50–69, 1988. ISSN 00219991.

D. Fultz. Studies in Experimental Hydrodynamics (1). *Final Report, Hydrodynamics Lab., University of Chicago*, 1956.

D. Gidaspow. *Multiphase flow and fluidization: continuum and kinetic theory descriptions*. Academic press, 1994.

C. Green. Improved alpha-tested magnification for vector textures and special effects. *ACM SIGGRAPH 2007 courses on - SIGGRAPH '07*, page 9, aug 2007.

S. Gsell, T. Bonometti, and D. Astruc. A coupled volume-of-fluid / immersed-boundary method for the study of propagating waves over complex-shaped bottom : Application to the solitary wave. *Computers and Fluids*, 131:56–65, 2016. ISSN 0045-7930.

E. Halley. An Historical Account of the Trade Winds, and Monsoons, Observable in the Seas between and Near the Tropicks, with an Attempt to Assign the Phisical Cause of the Said Winds, By E. Halley. *Philosophical Transactions of the Royal Society of London*, 16(179-191):153–168, jan 1686. ISSN 0261-0523.

E. J. Hinch and G. Schubert. Strong streaming induced by a moving thermal wave. *Journal of Fluid Mechanics*, 47(02):291, mar 1971. ISSN 0022-1120.

M. Hou, Z. Peng, R. Liu, K. Lu, and C. K. Chan. Dynamics of a projectile penetrating in granular systems. *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics*, 72(6), 2005. ISSN 15393755.

H. M. Jaeger, S. R. Nagel, and R. P. Behringer. The physics of granular materials. *Physics today*, 49:32–39, 1996.

A. K. Jain. *Fundamentals of digital image processing*. Prentice-Hall, Inc., 1989.

H. Jasak. *Error Analysis and Estimation for the Finite Volume Method with Applications to Fluid Flows*. PhD thesis, 1996.

H. Jasak, A. Jemcov, and Z. Tukovic. OpenFOAM : A C ++ Library for Complex Physics Simulations. *International Workshop on Coupled Methods in Numerical Dynamics*, m: 1–20, 2007.

C. Johnson and J. Gray. Granular jets and hydraulic jumps on an inclined plane. *Journal of Fluid Mechanics*, 675:87–116, 2011.

L. P. Kadanoff. Built upon sand: Theoretical ideas inspired by granular flows. *Reviews of Modern Physics*, 71(1):435, 1999.

T. Kajishima and S. Takiguchi. Interaction between particle clusters and particle-induced turbulence. *International Journal of Heat and Fluid Flow*, 23(5):639–646, 2002. ISSN 0142727X.

U. Lacis, N. Brosse, F. Ingremeau, A. Mazzino, F. Lundell, H. Kellay, and S. Bagheri. Passive appendages generate drift through symmetry breaking. *Nature communications*, 5:5310, 2014. ISSN 2041-1723.

M.-C. Lai and C. S. Peskin. An Immersed Boundary Method with Formal Second-Order Accuracy and Reduced Numerical Viscosity. *Journal of Computational Physics*, 160(2): 705–719, 2000. ISSN 00219991.

E. Lajeunesse. Spreading of a granular mass on a horizontal plane. 16(7), 2004.

E. Lajeunesse, A. Mangeney-Castelnau, and J. Vilotte. Spreading of a granular mass on a horizontal plane. *Physics of Fluids*, 16(7):2371–2381, 2004.

D. J. Laser and J. G. Santiago. A review of micropumps. *Journal of Micromechanics and Microengineering*, 14(6):R35–R64, jun 2004. ISSN 0960-1317.

C. Liao, Y. Chang, C. Lin, and J. McDonough. Simulating flows with moving rigid boundary using immersed-boundary method. *Computers & Fluids*, 39(1):152–167, jan 2010. ISSN 00457930.

D. Lohse, R. Bergmann, R. Mikkelsen, C. Zeilstra, D. van der Meer, M. Versluis, K. van der Weele, M. van der Hoef, and H. Kuipers. Impact on Soft Sand: Void Collapse and Jet Formation. *Physical Review Letters*, 93(19):198003, nov 2004. ISSN 0031-9007.

F. E. Loranca-Ramos, J. L. Carrillo-Estrada, and F. Pacheco-Vazquez. Craters and Granular Jets Generated by Underground Cavity Collapse. *Physical Review Letters*, 115(2): 1–5, 2015. ISSN 10797114.

W. E. Lorensen and H. E. Cline. Marching cubes: A high resolution 3D surface construction algorithm. In *Proceedings of the 14th annual conference on Computer graphics and interactive techniques - SIGGRAPH '87*, pages 163–169, 1987. ISBN 0897912276.

W. Mao, A. Oron, and A. Alexeev. Fluid transport in thin liquid films using traveling thermal waves. *Physics of Fluids*, 25(7):072101, 2013. ISSN 10706631.

J. Marston, E. Li, and S. T. Thoroddsen. Evolution of fluid-like granular ejecta generated by sphere impact. *Journal of Fluid Mechanics*, 704:5–36, 2012.

J. M. Mihaljan. A rigorous exposition of the boussinesq approximations applicable to a thin layer of fluid. *The Astrophysical Journal*, 136:1126, 1962.

R. Mikkelsen, M. Versluis, E. Koene, G.-W. Bruggert, D. van der Meer, K. van der Weele, and D. Lohse. Granular Eruptions: Void Collapse and Jet Formation. *Physics of Fluids*, 14(9):S14–S14, sep 2002. ISSN 1070-6631.

R. Mittal and G. Iaccarino. Immersed Boundary Methods. *Annual Review of Fluid Mechanics*, 37(1):239–261, jan 2005. ISSN 0066-4189.

J. Monaghan. Smoothed Particle Hydrodynamics and Its Diverse Applications. *Annual Review of Fluid Mechanics*, 44(1):323–346, 2012. ISSN 0066-4189.

R. Muthuraj and S. Srinivas. Mixed convective heat and mass transfer in a vertical wavy channel with traveling thermal waves and porous medium. *Computers & Mathematics with Applications*, 59(11):3516–3528, jun 2010. ISSN 08981221.

X. Nie, M. O. Robbins, and S. Chen. Resolving singular forces in cavity flow: Multiscale modeling from atomic to millimeter scales. *Physical Review Letters*, 96(13):2–5, 2006. ISSN 00319007.

P. C. Carman. Flow of Gases Through Porous Media. *Combustion and Flame*, 1(1):187, 1957. ISSN 00102180.

F. Pacheco-Vazquez, G. A. Caballero-Robledo, J. M. Solano-Altamirano, E. Altshuler, A. J. Batista-Leyva, and J. C. Ruiz-Suarez. Infinite penetration of a projectile into a granular medium. *Physical Review Letters*, 106(21), 2011. ISSN 00319007.

C. S. Peskin. Numerical analysis of blood flow in the heart, 1977. ISSN 00219991.

F. Radjai, S. Roux, and J. J. Moreau. Contact forces in a granular packing. *Chaos (Woodbury, N.Y.)*, 9(3):544–550, 1999. ISSN 1089-7682.

A. Rahman. Correlations in the Motions of Atoms in Liquid Argon. *Physical Review*, 136 (2A):A405—-A411, 1964. ISSN 0031-899X.

D. R. Reyes, D. Iossifidis, P. A. Auroux, and A. Manz. Micro total analysis systems. 1. Introduction, theory, and technology. *Analytical Chemistry*, 74(12):2623–2636, 2002. ISSN 0003-2700.

O. Reynolds. Experiments showing dilatancy, a property of granular material, possibly connected with gravitation. *Proc. R. Inst. GB*, 1886.

P. Richard, M. Nicodemi, R. Delannay, P. Ribière, and D. Bideau. Slow relaxation and compaction of granular systems. *Nature materials*, 4(2):121–128, 2005. ISSN 1476-1122.

G. H. Ristow. *Pattern formation in granular materials*. Number 164. Springer Science & Business Media, 2000.

J. R. Royer, E. I. Corwin, A. Flior, M.-L. Cordero, M. L. Rivers, P. J. Eng, and H. M. Jaeger. Formation of granular jets observed by high-speed X-ray radiography. *Nature Physics*, 1(3):164–167, 2005. ISSN 1476-0000.

J. R. Royer, E. I. Corwin, B. Conyers, A. Flior, M. L. Rivers, P. J. Eng, and H. M. Jaeger. Birth and growth of a granular jet. *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics*, 78(1), 2008. ISSN 15393755.

G. Schubert and J. Whitehead. Moving flame experiment with liquid mercury: Possible implications for the Venus atmosphere. *Science*, 163(3862):71–72, 1969.

G. Schubert, R. Young, and J. Hinch. Prograde and retrograde motion in a fluid layer: Consequences for thermal diffusion in the Venus atmosphere. *Journal of Geophysical . . .*, 76(9):2126–2130, 1971.

J. Shäfer, S. Dippel, and D. Wolf. Force schemes in simulations of granular materials. *Journal de physique I*, 6(1):5–20, 1996.

P. Shankar and M. Deshpande. Fluid mechanics in the driven cavity. *Annual Review of Fluid Mechanics*, 32:93–136, 2000. ISSN 0066-4189.

N. Sharma and N. A. Patankar. A fast computation technique for the direct numerical simulation of rigid particulate flows. *Journal of Computational Physics*, 205(2):439–457, 2005. ISSN 00219991.

J. Shen. Hopf bifurcation of the unsteady regularized driven cavity flow. *Journal of Computational Physics*, 95(1):228–245, 1991. ISSN 10902716.

Y. Shi, G. H. Tang, and H. H. Xia. Lattice Boltzmann simulation of droplet formation in T-junction and flow focusing devices. *Computers and Fluids*, 90:155–163, 2014. ISSN 00457930.

L. Staron and E. J. Hinch. Study of the collapse of granular columns using DEM numerical simulation. *Journal of Fluid Mechanics*, 545:1, 2005. ISSN 0022-1120.

M. Stern. The moving flame experiment. *Tellus*, 1959.

H. A. Stone, A. D. Stroock, and A. Ajdari. Engineering flows in small devices: microfluidics toward a lab-on-a-chip. *Annual Review of Fluid Mechanics*, 36(1):381–411, 2004. ISSN 0066-4189.

X. Sun and M. Sakai. Three-dimensional simulation of gassolidliquid flows using the DEMVOF method. *Chemical Engineering Science*, 134:531–548, 2015. ISSN 00092509.

A. ten Cate, C. H. Nieuwstad, J. J. Derksen, and H. E. a. Van den Akker. Particle imaging velocimetry experiments and lattice-Botlzmann simulations on a single sphere settling under gravity. *Physics of Fluids*, 14(11):4012–4025, 2002. ISSN 10706631.

K. Terzaghi. *Theoretical soil mechanics*, volume 18. Wiley Online Library, 1943.

M. Teschner, S. Kimmerle, B. Heidelberger, G. Zachmann, L. Raghupathi, A. Fuhrman, M.-p. Cani, N. Magnenat-thalmann, W. Strasser, M. Teschner, S. Kimmerle, B. Heidelberger, G. Zachmann, and L. Raghu. Collision Detection for Deformable Objects To cite this version : Collision Detection for Deformable Objects. 2005.

S. T. Thoroddsen and A. Q. Shen. Granular jets. *Physics of Fluids*, 13(1):4–6, 2001. ISSN 10706631.

Y. Tseng and J. H. Ferziger. A ghost-cell immersed boundary method for flow in complex geometry. *Journal of Computational Physics*, 192(2):593–623, dec 2003. ISSN 00219991.

Y. Tsuji, T. Tanaka, and T. Ishida. Lagrangian numerical simulation of plug flow of cohesionless particles in a horizontal pipe. *Powder technology*, 71(3):239–250, 1992.

M. Uhlmann. An immersed boundary method with direct forcing for the simulation of particulate flows. *Journal of Computational Physics*, 209(2):448–476, nov 2005. ISSN 00219991.

F. Weinert, J. Kraus, T. Franosch, and D. Braun. Microscale Fluid Flow Induced by Thermoviscous Expansion Along a Traveling Wave. *Physical Review Letters*, 100(16): 164501, apr 2008. ISSN 0031-9007.

C. Wu, A. S. Berrouk, and K. Nandakumar. Three-dimensional discrete particle model for gassolid fluidized beds on unstructured mesh. *Chemical Engineering Journal*, 152(2-3): 514–529, oct 2009. ISSN 13858947.

C. Wu, O. Ayeni, A. S. Berrouk, and K. Nandakumar. Parallel algorithms for CFDDEM modeling of dense particulate flows. *Chemical Engineering ...*, 2014.

Y. Xu, J. Padding, M. van der Hoef, and J. Kuipers. Detailed numerical simulation of an intruder impacting on a granular bed using a hybrid discrete particle and immersed boundary (dp-ib) method. *Chemical engineering science*, 104:201–207, 2013.

E. Yariv and H. Brenner. Flow animation by unsteady temperature fields. *Physics of Fluids*, 16(11):L95, 2004. ISSN 10706631.

T. Ye, R. Mittal, H. Udaykumar, and W. Shyy. An Accurate Cartesian Grid Method for Viscous Incompressible Flows with Complex Immersed Boundaries. *Journal of Computational Physics*, 156(2):209–240, dec 1999. ISSN 00219991.

Y. Yuki, S. Takeuchi, and T. Kajishima. Efficient Immersed Boundary Method for Strong Interaction Problem of Arbitrary Shape Object with the Self-Induced Flow. *Journal of Fluid Science and Technology*, 2(1):1–11, 2007. ISSN 1880-5558.

Appendix: Error Estimation in Presence of Solution Singularity

A.1 Introduction

Accuracy is a key feature of all numerical algorithms. Every numerical algorithm has one description stating "it is$n$-th order accurate". The order often refers to the truncation error. It is more subjective to improvement compared with rounding off error, and it provides valuable information for practical questions like how fine or finer a mesh (or time step) needs to be. In this appendix we use *order* to refer to the order of the truncation error, which often determines the order of the overall numerical algorithm.

Truncation error is a byproduct of the derivation of numerical algorithms. It has the general form $c\delta^n$: c is an unknown constant, δ is the discretization resolution (time step or grid size), and n is the order. This expression is informative, but it is always necessary to prove the order with numerical experiment. The common way of numerically evaluating the order is by conducting several numerical experiments—each differs only in δ value, measuring the error using L^1 or L^2 norm, and fitting an error-versus-resolution line in a log-log plot; the slope of the fitting line gives the order. We will refer to this process as *order estimation*.

We are concerned with spatial order. We show through several numerical experiments that the singularities in the solution cause underestimations of the order if they are included when computing the error norm. The singularity is not necessarily developed from the system's physics, but could be introduced through boundary conditions to smooth linear operators. The next Section presents three numerical experiments. We start with the two-dimensional lid driven cavity problem, followed by two-dimensional Laplace equation, and by one-dimensional numerical integration. They are arranged towards simpler algorithm to emphasize that the problem with order estimation is not caused by the convergence of nonlinear/linear solvers, but is general for numerical algorithms with different levels of complexity. Section 4.3 presents a theoretical model of order estimation in presence of a $1/x$ type singularity. Section 4.4 concludes this study.

A.2 Numerical Experiments

A.2.1 Two Dimensional Lid Driven Cavity

Two dimensional lid driven cavity (hereafter referred as the cavity problem) is a well studied classic fluid dynamics problem [Shankar and Deshpande, 2000]. It is popular in computational fluid dynamics and is a common benchmark of algorithm order and implementation. Figure A.1 shows the domain, the governing incompressible Navier-Stokes Equation, the boundary conditions, and the definition of symbol d used later in this subsection.

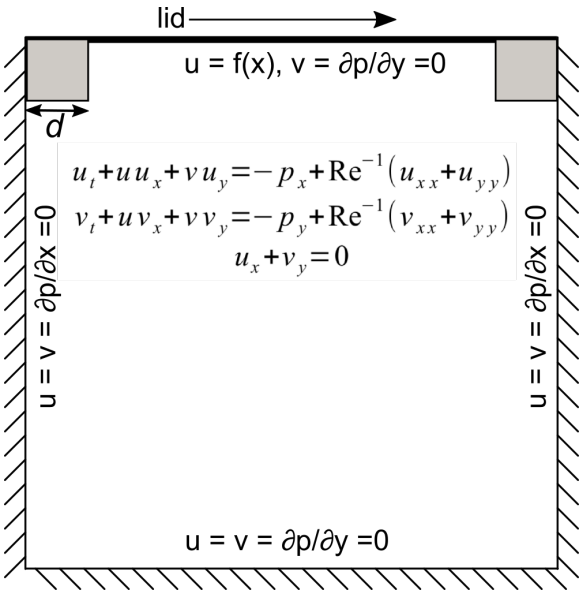

Figure A.1: Illustration of the driven cavity problem; also shown are the governing equation, boundary conditions, and the definition of d used later.

Table A.1: Summary of options for the cavity problem

Reynolds number	1	10	100
Cavity type	Singular		Regular
Test Grid Resolution	16×16	32×32	64×64
Advection scheme order	Linear (second order)		Upwind (first order)

In the classic cavity problem, the lid velocity $f(x)$ is a constant. This means the u-velocity boundary conditions at the top two corners are discontinuous. This poses a direct challenge to certain numerical algorithm like finite difference, which has two grid nodes laying exactly on the corner, making it impossible to specify boundary conditions there. But finite volume method can avoid this issue by taking a control volume point of view. Regarding the physics, it was stated that [Shankar and Deshpande, 2000] "*the singularities have virtually no effect over most of the flow field, their effects being confined to the neighborhood of the singular corners*". This is true for most uses of this problem. But those corners do cause problems if they are included during order estimation.

Our numerical experiment covers a wide range of options to ensure a general observation (Table 1). The Reynolds number, which is the problem's only physical parameter, is varied by two orders of magnitude. For comparison, we use both the classic cavity discussed above (will be referred as singular cavity), and the "regular cavity" [Shen, 1991]. The latter differs from the singular cavity only in the lid velocity $f(x) = 4x(1-x)$. The SIMPLE algorithm implemented by OpenFOAM [Jasak et al., 2007] version 3.0 is used. We control the algorithm order through the advection term. The upwind scheme is used to achieve overall first order accuracy because the solution error is dominated by the lowest order discretization. The linear scheme is used to have a second order accurate algorithm. The viscous term uses the standard second order scheme in all cases.

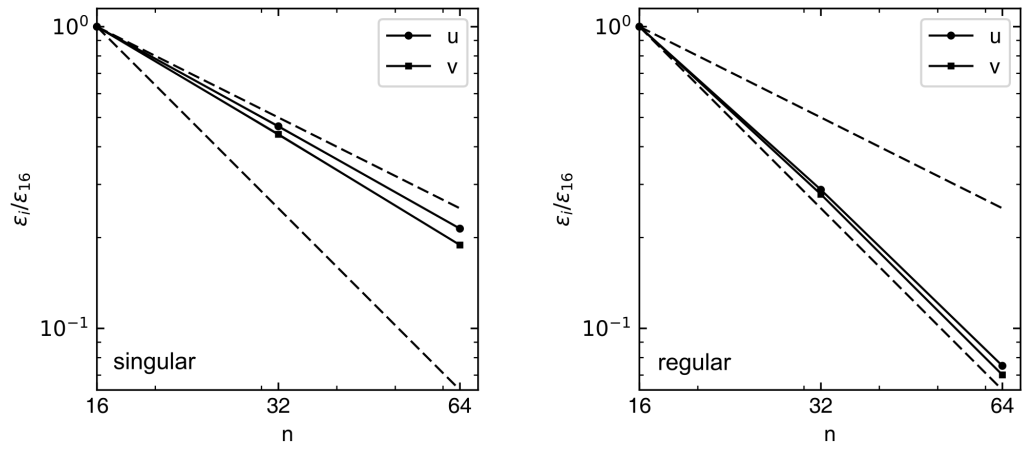

Figure A.2: Two example order estimations. Left: singular cavity at Re = 100; right: regular cavity at Re = 100. Both are calculated with the second order algorithm. The two dashed lines represent the first and second order accuracy.

For the purpose of order estimation, we obtain *test solutions* from three coarse test grids—there are $3 \times 2 \times 3 \times 2 = 36$ of them. For each different flow pattern determined by Reynolds number and cavity type (3×2 combinations, from row 1 and 2 of Table 1), a *reference solution* is obtained on a 200×200 grid with SIMPLE algorithm and second order scheme. The reference solution is obtained first. It is then linearly interpolated onto the test grids to initialize the flow fields for the test solutions. This enforces "accurate" initial conditions for all test solutions, such that any final error is caused by grid resolution and algorithm order. Solutions are considered converged when both (u, v) have residue smaller than 10^{-9}.

The difference between the reference and test solutions is used to calculate the error norm ε, the test solution is linearly interpolated onto the reference grid for this purpose. The linear interpolation is second order accurate and does not bias the order estimation. Using L^2 norm:

$$\epsilon = \sqrt{\frac{1}{200 \times 200} \sum_{m=1}^{200} \sum_{n=1}^{200} (\tilde{\varphi}_{m,n} - \varphi_{m,n})^2}$$

φ is u or v, the tilde means the test solution. L^1 norm was tested and gave similar observations, so we limit to L^2 norm. There are three ε values from the three test grids. A linear fit of $\varepsilon - n$ in a $\log - \log$ plot gives the estimated order. Two of such results are shown in figure A.2, other results (regardless of Reynolds number) share similar properties and are omitted from graphical presentation. The key observation is when the second order scheme is used, the order is underestimated using the singular cavity. Regular cavity, on the other hand, has no such problem.

When the first order algorithm is used, both cavities give the correct order estimation (not shown). The reason is likely due to that the first order error dominates over any other errors. Since schemes with only first order spatial accuracy are usually unacceptable for practical applications, we do not discuss it further. When the second order algorithm is used, the problem with order estimation is apparent with the singular cavity. The fact that the regular cavity always gives order estimation close to two is a strong indication that the singularity in the boundary conditions, being the main difference between the two cavities, is the cause. This is confirmed by figure A.3, which shows the effects of excluding different sized square regions from the error norm—thus from order estimation. Each square has one corner coinciding with the domain's (top) corner, and encloses regions around the singular corners. Figure A.3 shows that by excluding sufficiently large regions around the singular corners, the order estimation recovers the expected value of two.

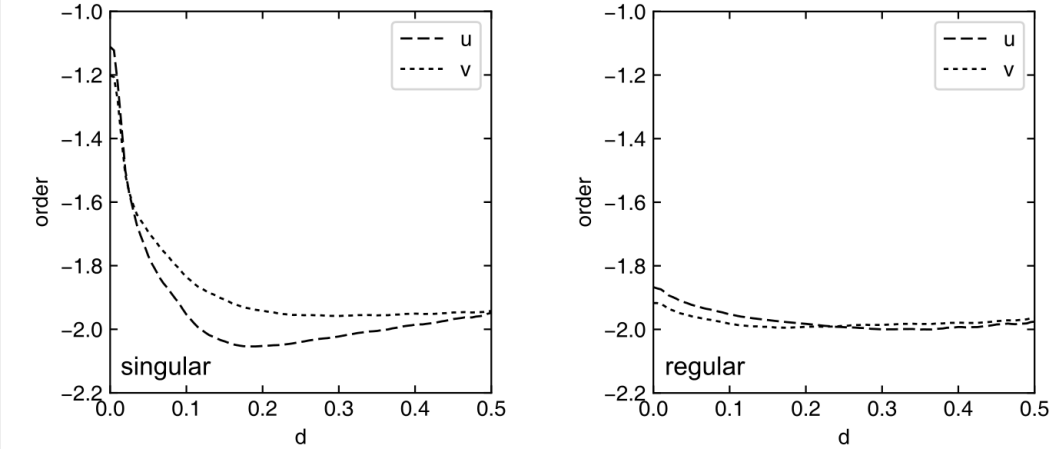

Figure A.3: Effect of excluding the singular corners from order estimation for the cavity problem. d in the x-axis is the size of the shaded square shown in figure A.1. The left panel corresponds to the singular cavity. It is hard to tell that the scheme is second order when d is smaller than, say, 0.05. The right panel corresponds to the regular cavity and does not have this issue.

A.2.2 Two Dimensional Laplace Equation

The Laplace equation $\Delta T = 0$ models, among others, the steady state heat diffusion. We use the same (both test and reference) grids as that were used in the cavity problem. The left and bottom boundary conditions are symmetric. The right wall is applied $T = 0$ condition. The top wall (lid) is applied $T = 1$ condition to create a "singular thermal cavity", and applied $T = 1 - x^2$ condition to create a "regular thermal cavity". The second order accurate laplacianFoam from OpenFOAM 3.0 is used for solution. The convergence criterion is residue $\leq 10^{-12}$.

The singularity causes underestimation of the algorithm order in much the same way as it does for the cavity problem. Similarly, the regular thermal cavity has no such problem. Those observations are summarized in figure A.4. This experiment with Laplace equation

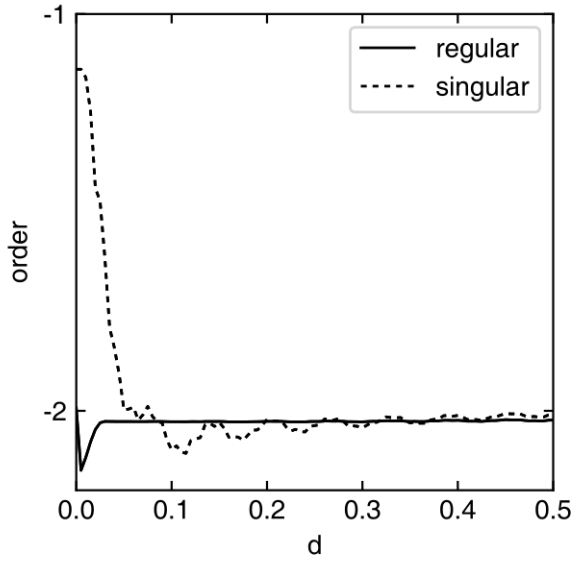

Figure A.4: Effect of excluding the singular corners from order estimation, for the Laplace equation. d is the size of the shaded square shown in figure A.1: only the top right corner is excluded, as the top left corner is not singular.

suggests that the problem with order estimation in the cavity problem is not related with the nonlinear governing equations. Instead, it is caused by the singular boundary conditions.

A.3 One Dimensional Integration

In this subsection we study the integration:

$$I = \int_s^{s+1} x^{-m} dx$$

The integrand x^{-m} has different strengths of singularity by controlled by m, m being a positive integer. The s (for shift) in the integration bound is a non-negative real number. By increasing s we move away from the singularity at the origin, and can expect the effect of the singularity to diminish. The rectangle rule is used. It uses integrand values at interval centers, rather than at the interval bounds like the trapezoidal rule. This way it is similar to the previous finite volume calculations where all unknowns (and their errors) locate at cell centers. It is well-known that the rectangle rule is second order accurate. This problem is very rudimentary. It certainly has no dynamical behaviors like fluid dynamics or heat diffusion, but the underestimation of the algorithm order caused by singularities persists.

Uniform grids with resolution $n = 16, 32, 64, 128$, and 256 are used to generate test integrations. A fine grid with $n = 1024$ is used to generate the reference integration. We do have analytical solutions when $s \neq 0$, but using it instead has minor effects on the order estimation, so we stay with numerical solutions for consistency.

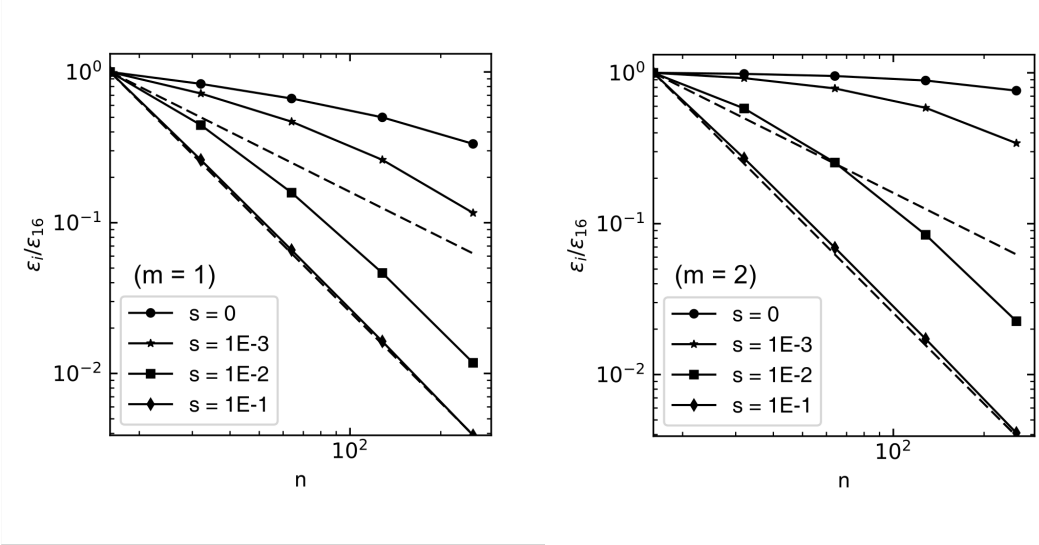

Figure A.5: Effects of interval shift s for the case $m = 1$ (left) and $m = 2$ (right). Each solid line corresponds to an s value, and it draws five normalized errors from different test grids ($n = 16, 32, 64, 128, 256$). The normalized error is the error norm normalized by ϵ_{16}, ϵ_{16} being the error norm of the coarsest $n = 16$ grid. The two dashed lines represent the first and second order.

Figure A.5 summarizes the effects of s on the order estimation. Two functions, x^{-1} and x^{-2}, are used. The correct second order behavior is captured only when the interval is sufficiently away from the singularity (larger s), much like the regions excluded in experiment 1 and 2 have to be large enough to recover the second order estimation. In the left panel for x^{-1}: when $s = 10^{-1}$, the estimated order is correct just with the two coarsest grids (16 and 32); when $s = 10^{-2}$ the 16 and 32 grids do not have second order behavior, and grids with resolution $n \geq 64$ are needed; when $s = 10^{-3}$ it appears the two finest grids are approaching the second order behavior; finally, when $s = 0$, even with the finest grids the estimated order is less than one. The x^{-2} function has stronger singularity, and the order estimation is worse systematically except when $s = 0.1$.

The reason that the error in the integration grows with s can be explained by noting that the upper error bound of the rectangle rule, applied to a general function $f(x)$ in $[a, b]$, is

$$|\epsilon| \leq \frac{(b-a)^3}{24N^2}|f''(\xi)| = \frac{m(m+1)}{24s^{m+2}N^2}$$

N is the number of uniform interval subdivisions. The term after the equal sign is the bound applied to our case. To limit the error within a prescribed bound requires $s^{m+2}N^2$ stay constant, or $N \sim s^{(-m/2-1)}$. In other words, a finer grid is necessary to bound the error as the singularity is approached. In addition, a stronger singularity (larger m) requires more grid points. This however does not explain what happens when $s = 0$ because the error bound is then infinity. The $s = 0$ curve in figure A.5 appears to have better order estimation as n increases, but it won't converge to the correct value of two, as will be shown in the next section.

A.4 The $1/x$ Singularity

We consider the rectangle rule applied to $1/x$ in $[0,1]$ in detail, and show there is a theoretical upper limit of the estimated order using the common "grid resolution doubling" strategy. The analysis of this simple singularity in the simplest numerical operation provides valuable insights to questions like the source of the underestimation, and the possibility of containing the singularity effect by local grid refinement.

Suppose we have a set of successively finer grids on $[0,1]$, with grid cell count $N(n) = 2^{n-1}N_0$. The $N(1)$ grid is the coarsest grid with N_0 cells, and each finer grid has a doubled resolution. Then for any second order algorithm, the error ϵ has the functional form:

$$\varepsilon = \frac{C}{N^2} = \frac{4C}{N_0^2 2^{2n}}$$

C is some constant and is not important here. The algorithm's order is $-\partial log 2\epsilon/\partial n$. Let the $N(1)$ grid's cell size $1/N_0 = \delta$, each grid refinement divides an $N(1)$ grid cell into smaller cells. We separate all N cells of the $N(n)$ grid into two groups: group A with 2^{n-1} cells covering interval $[0,\delta]$; and group B with $N - 2^{n-1}$cells covering the remaining interval $[\delta,1]$.

Group A has a singularity at its left bound. On the $N(1)$ grid, group A has one cell and the integration I_A equals $f(\delta/2)\delta$. On a general $N(n)$ grid I_A is a sum of 2^{n-1} items:

$$I_A(n) = \frac{\delta}{n} \cdot \sum_{k=1}^{2^{n-1}} f(\frac{2k-1}{2 \cdot 2^{n-1}}\delta) = \psi_0(2^{n-1} + \frac{1}{2}) - \psi_0(\frac{1}{2})$$

ψ_0 is the zeroth order poly-Γ function defined as $d(\ln\Gamma(x))/dx$, where $\Gamma(x)$ is the Gamma function. If the reference grid (the one used to calculate the reference solution) has $n = n_\infty$, the error of I_A is:

$$\epsilon_A = \psi_0(2^{n_\infty - 1} + \frac{1}{2}) - \psi_0(2^{n-1} + \frac{1}{2})$$

It is interesting that δ does not appear in equation (1A) (it appears when other power functions like $1/x^2$ are used). This means equation (1A) can be used to find the analytical expression of the whole integration in $[0,1]$. Let $\delta = 1$ and replace n with N gives the result:

$$I(n) = \psi_0(N + \frac{1}{2}) - \psi_0(\frac{1}{2})$$

As expected, $I(n)$ diverges as $N \to \infty$. I_B, the numerical integration over group B, equals $I - I_A$:

$$I_B(n) = \psi_0(N + \frac{1}{2}) - \psi_0(2^{n-1} + \frac{1}{2}) = \psi_0(N + \frac{1}{2}) - \psi_0(\frac{N}{N_0} + \frac{1}{2})$$

To calculate the error contributed by I_B, we need a reference solution of I_B. We choose to use the analytical solution $\ln 1 - \ln(1/N_0) = \ln(N_0)$. The reason is interval

Table A.2: Integration error calculated numerically and predicted by (1C) for different grids

n	1	2	3	4	5
Grid size	16	32	64	128	256
Numerical	4.1587	3.4657	2.7726	2.0794	1.3863
Theoretical	4.1172	3.4553	2.7700	2.0788	1.3861

Table A.3: Order estimation calculated numerically and predicted by Equation (2) for different grid

n	1	2	3	4	5
Grid size	16	32	64	128	256
Numerical	0.2630	0.3219	0.4150	0.5850	0.5850
Theoretical	0.2226	0.2834	0.3591	0.4804	0.7212

$[\delta, 1]$ does not have singularity, and a numerical integration with a very fine grid $N(n_\infty)$ is very close to the analytical solution. Experiment confirmed that using either $\ln(N_0)$ or the $N(n_\infty)$ numerical solution here has negligible effects on order estimation. Apply the large z asymptotic expansion $\psi_0(z) = \ln(z) - 1/(2z) - 1/(12z^2) + O(z^{-3})$[Abramowitz and Stegun, 1964], the error contributed byI_B, to the leading order, is:

$$\epsilon_B = \frac{-N_0^2}{6(2N + N_0)^2} = \frac{-1}{6(2^n + 1)^2} \sim -\frac{1}{6} 2^{-2n} \tag{A.1}$$

ϵ_B has the expected second order functional form. Apply the same asymptotic expansion to ϵ_A and retain only the leading order terms, summing it with ϵ_B gives the final theoretical prediction of the total numerical error:

$$\epsilon = (\ln 2)(n_\infty - n) - \frac{1}{6} 2^{-2n} \tag{A.2}$$

Our numerical experiment has grids $n_\infty = 7, n = 1, 2, 3, 4, 5$, and $N_0 = 16$. The numerical and theoretical errors are listed in Table 4.2. It shows equation A.2 well captures the error, especially considering that we are only using the leading order asymptotics.

The theoretical order of accuracy $Or(n)$ is obtained with $-\partial \log 2\epsilon / \partial n$:

$$\mathrm{Or}(n) = \frac{3 \cdot 4^n - 1}{4^n (\ln 8)(n_\infty - n) - 1/2} \tag{A.3}$$

Note that without the singularity the result would be $Or(n) = 2$, a constant. The numerical order of accuracy (referring to figure A.5) is calculated with finite difference: forward difference for the first four n values, and backward difference for the last n value. The result is summarized in table 4.3. The differences between the theoretical and numerical orders are likely caused by the asymptotic approximation. But the overall agreement is sufficient to show that the singularity has great impact on the error and the order estimation.

So how good can the numerical order estimation be? In equation A.3, even with $n_\infty = 10$ and $n = 9$, which is much larger than common practices, the order is only about 1.44. If we keep $n_\infty - n$ to some integer A, it is easy to see that $Or(n)$ increases monotonically with n. In the limit of $n \to \infty$ the highest possible order is $3/(\ln 8A)$: for $A = 1$ it is about 1.4427, for $A = 2$ it is about 0.7213. So it is impossible to obtain the correct second order with the resolution doubling strategy. On the other hand, if a real valued $A < 1$ is used so the next finer grid is no longer a subdivision of the current grid, the order can be greater than two. In any case, the estimated order depends on the specific grid refinement strategy and is not reliable.

A.5 Discussion and Conclusion

Returning to the cavity problem in section 2, Batchelor [1967] shows the viscous stress on the lid diverges as $1/x$, where x is the distance between a location on the top lid and the nearby singular corner. The solution is in the Stokes regime but applies if we zoom sufficiently into the corner. So the discontinuity exists no matter how fine the grid is. From a simulation point of view, eventually the singularity has to be removed by molecular dynamics to reconcile with physical reality [Nie et al., 2006]. Figure A.6 shows the (absolute value of) error in u velocity for the singular and regular cavities. Even though the top lid boundary conditions are different, the two have similar patterns. In fact, in regions away from the top two corners the error fields appear quantitatively close. However, the error around the two singular corners is so large compared with rest of the domain that it, as we have shown, greatly affects the L^2 norm of the entire domain. In addition, the error convects to the inner domain. That is why in figure A.3 (left) not just the corner grid cell, but a larger region around it, needs to be excluded to recover the second order behavior. For this reason it is recommended to use the vertical (or horizontal) middle sections to sample values and evaluate errors for the cavity problem. This is a natural choice here due to the cavity geometry and is used by many. In other cases however, caution is needed to avoid singular regions.

The computational cases explored are quite general. Practical applications can have various kinds of singularity. For example, in fluid dynamics it is common for a model to have singularities caused by interfaces or boundaries. Our results suggest that regions around the singularity should be excluded when evaluating the accuracy of the numerical algorithm in use, for examples, by sampling data away from the singularity.

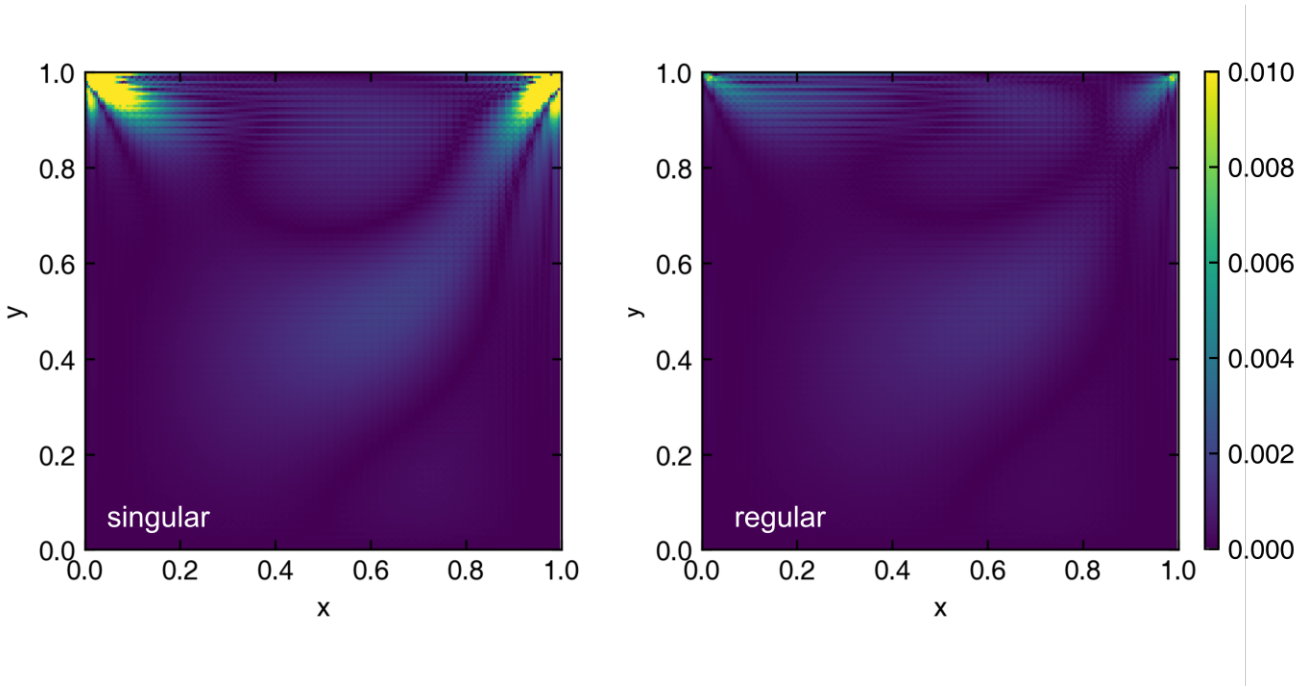

Figure A.6: Absolute value of the error in u-velocity field. Both cases have Re $= 100$ and grid resolution 64×64, For better comparison the two figures are clipped to the same value range from 0 to 0.01.

Vita

Chenguang Zhang graduated from Nanjing University, China in 2009, with a B.S. in physical geography. He then came to Louisiana State University to study physical oceanography under the guidance of Dr. Chunyan Li. After obtaining a M.S. in physical oceanography and coastal sciences, he joined Dr. Krishnaswamy Nandakumar's group and pursue a PhD in Engineering Science. At the same time a M.S. in computer science is finished, advised by Dr. Xin Li.